T0020 92207860

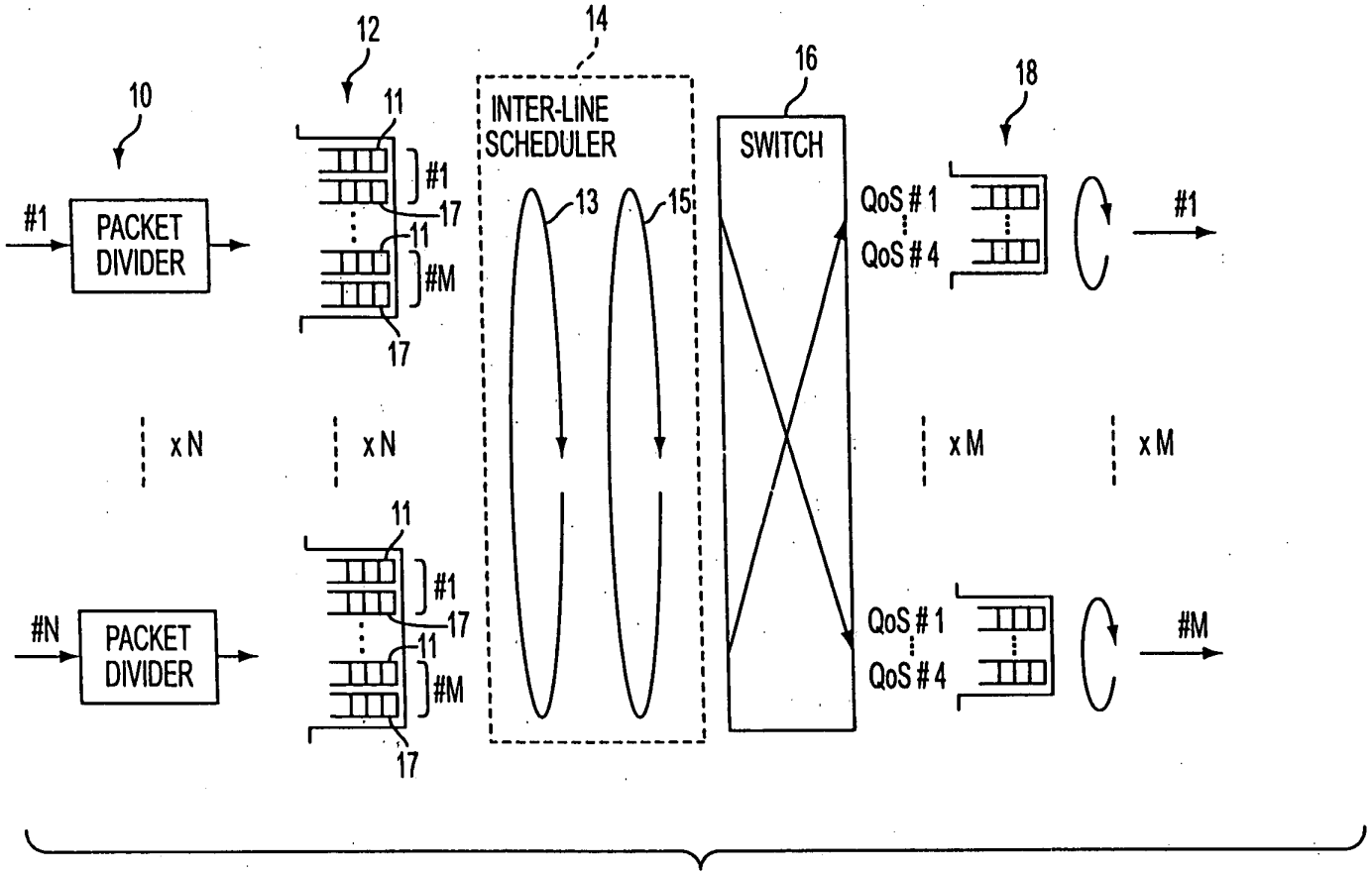


FIG. 1

T00260 92207860

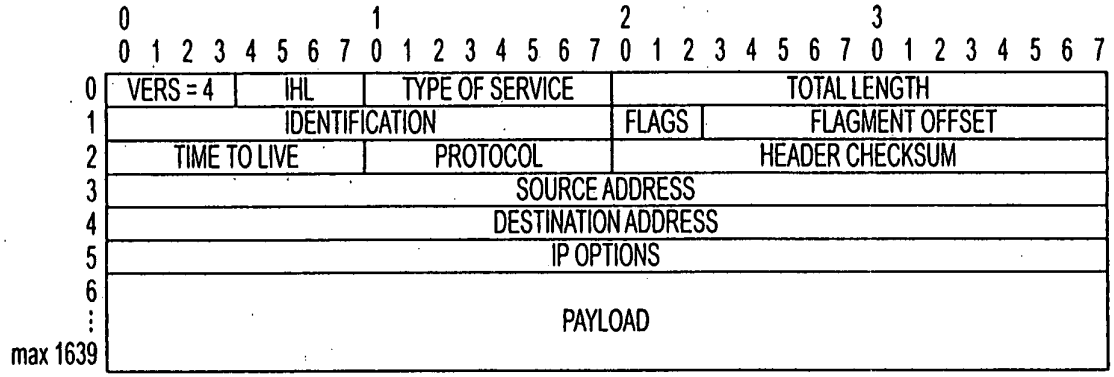


FIG. 2

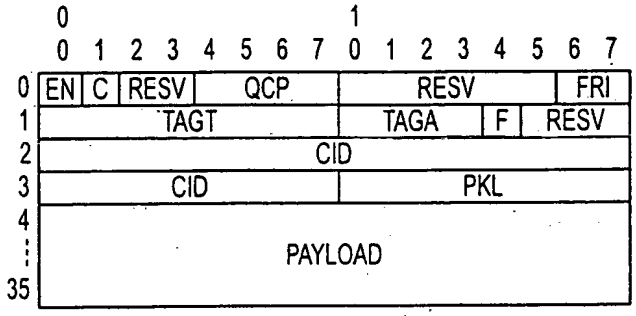

0 1
0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
0
1
2
3
4
35
EN
C
RESV
QCP
RESV
FRI
TAGT
TAGA
F
RESV
CID
CID
PKL
PAYLOAD

FIG. 3

T00260 "922E760

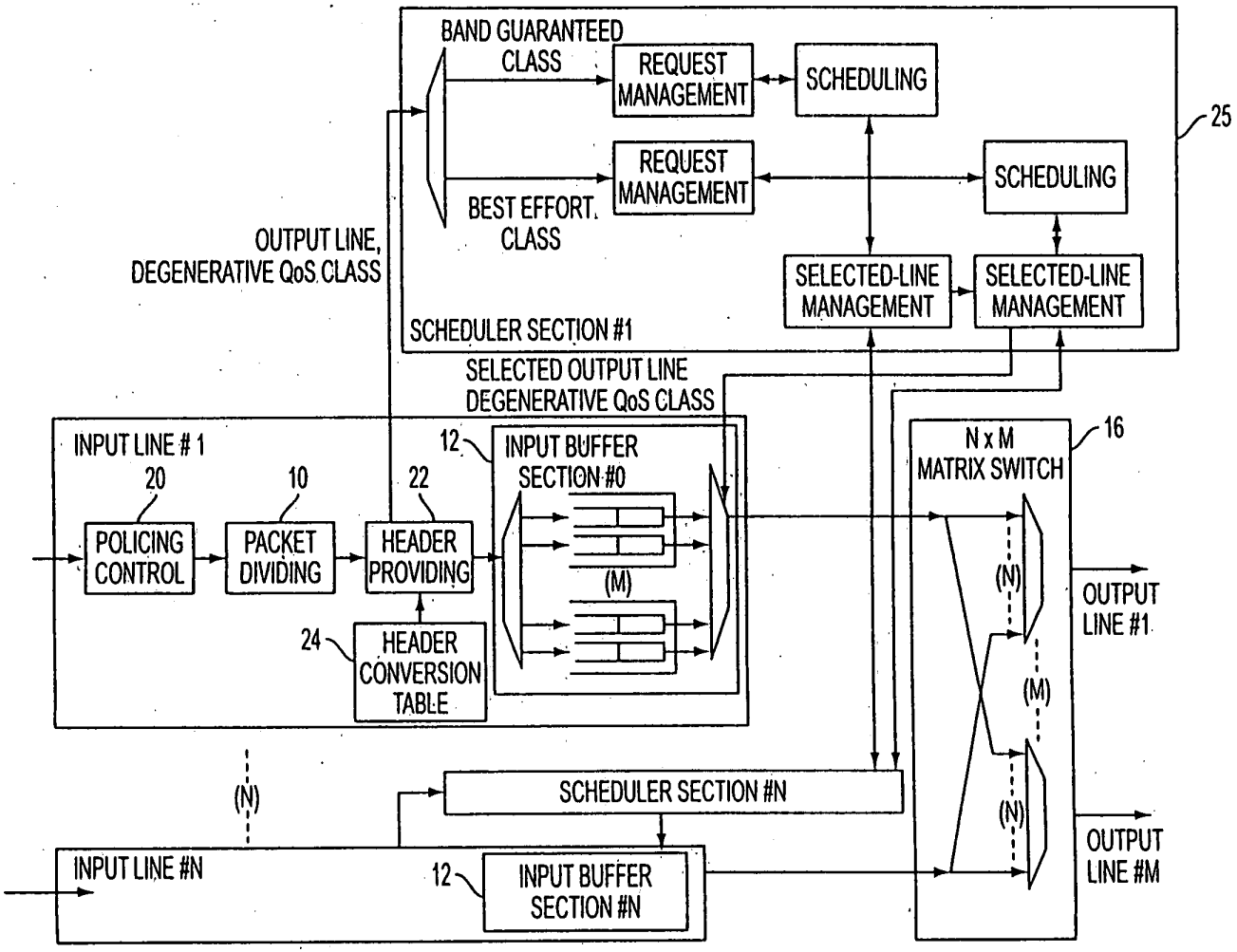


FIG. 4

T00220 "922ET860

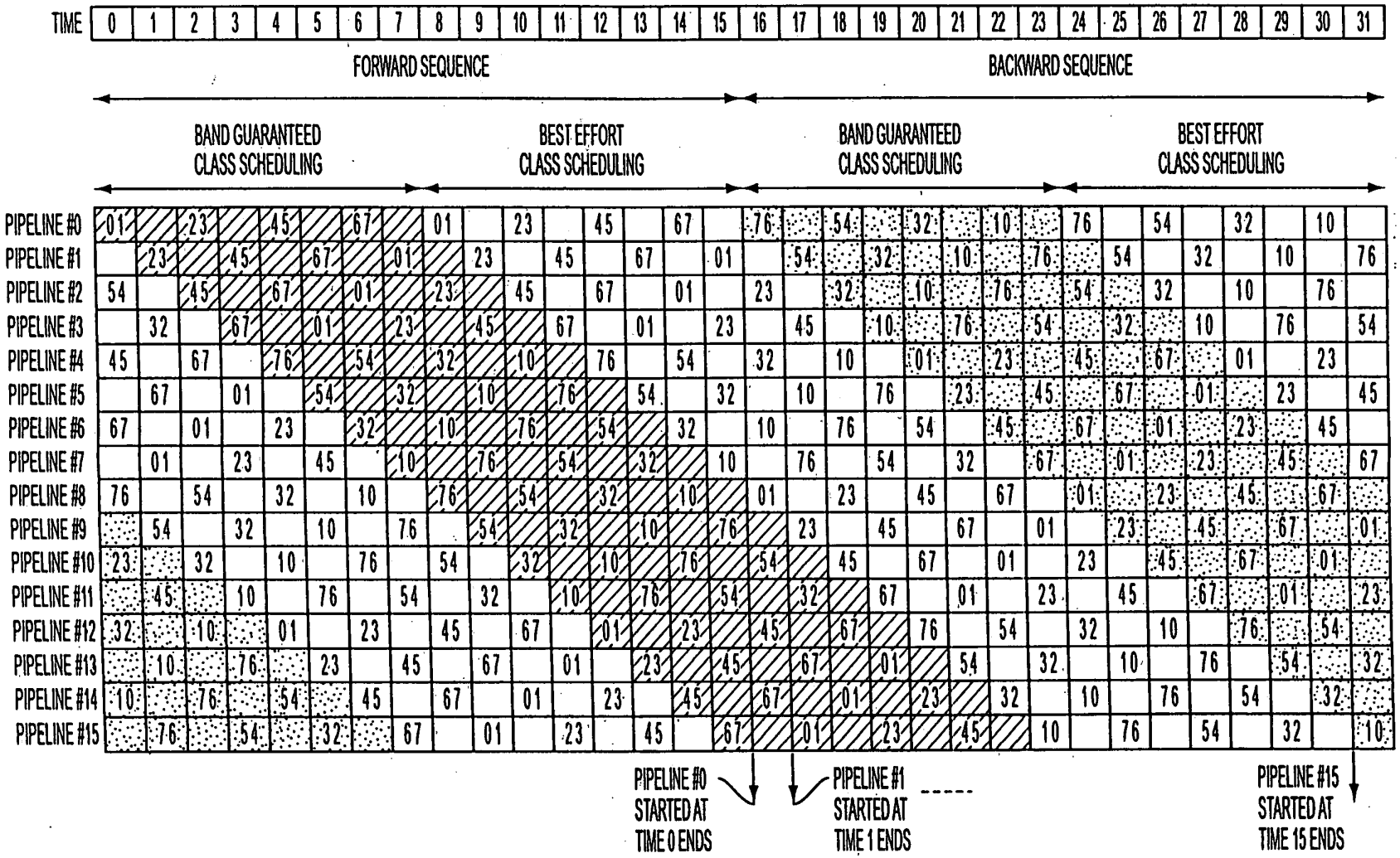


FIG. 5

T0020 "922T860

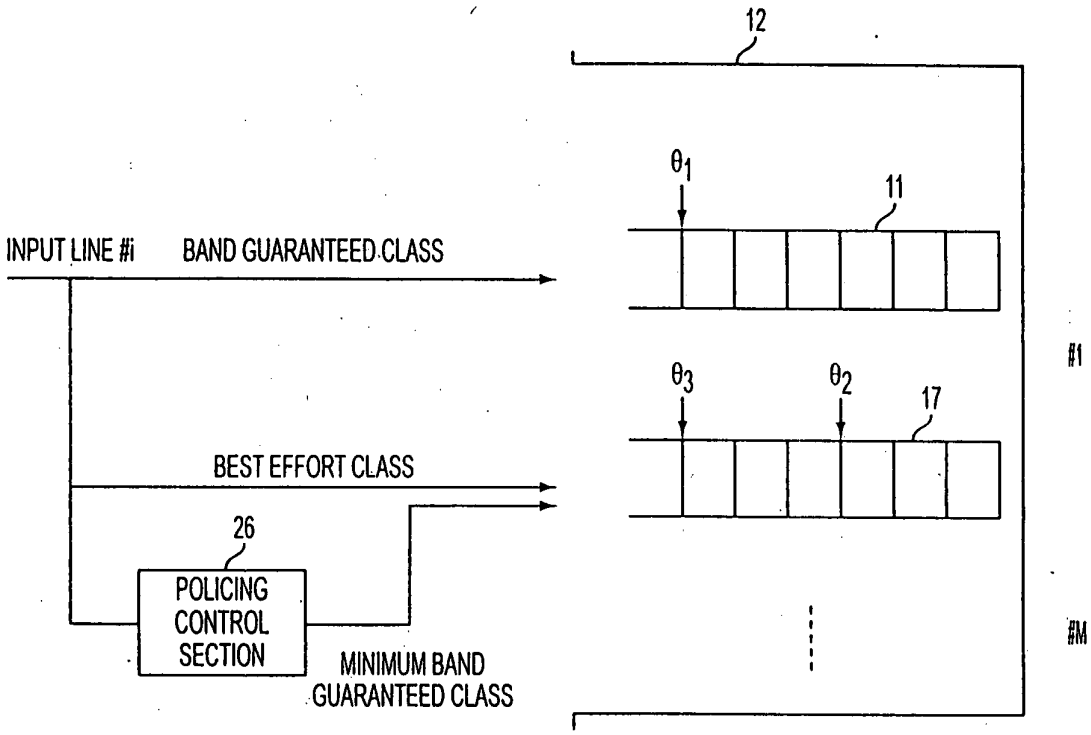


FIG. 6

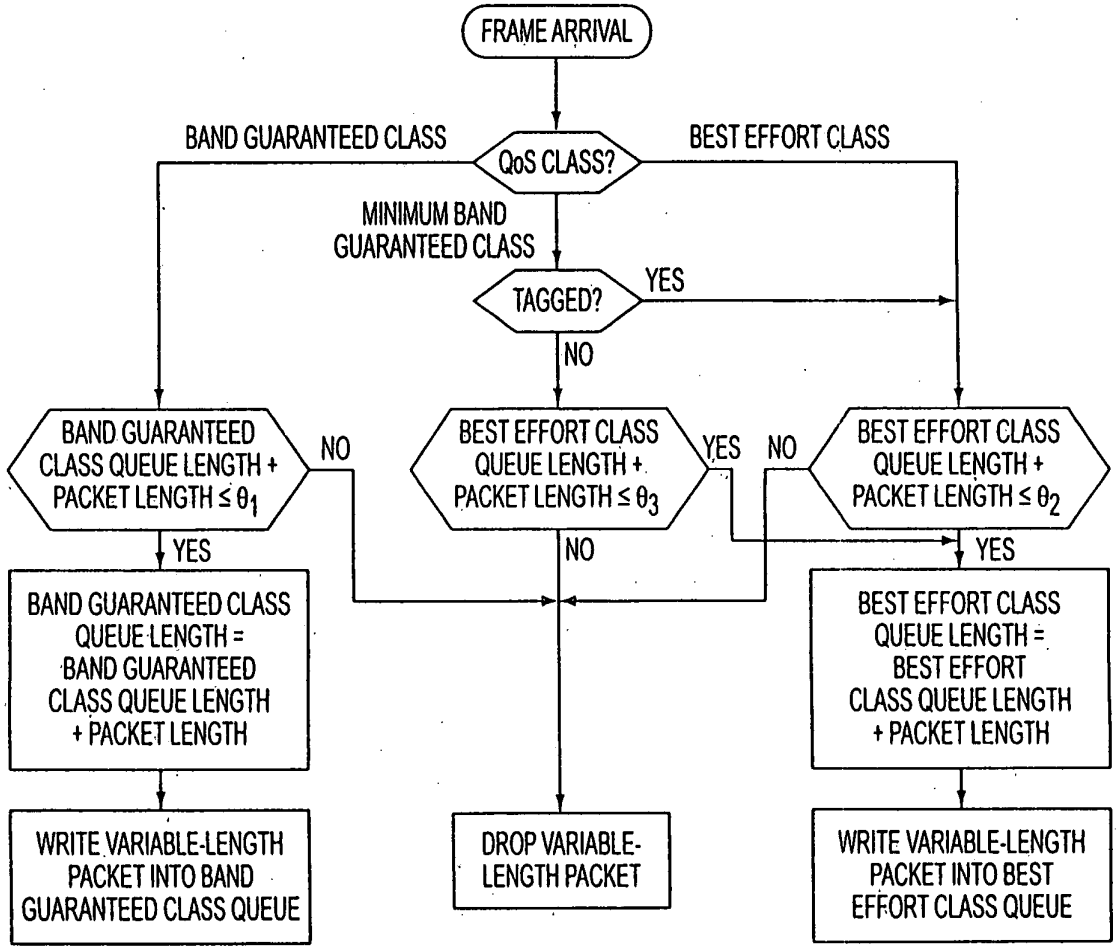

FRAME ARRIVAL
QoS CLASS?
BAND GUARANTEED CLASS
BEST EFFORT CLASS
MINIMUM BAND GUARANTEED CLASS
TAGGED?
YES
NO
BAND GUARANTEED CLASS QUEUE LENGTH + PACKET LENGTH ≤ θ1
NO
YES
BEST EFFORT CLASS QUEUE LENGTH + PACKET LENGTH ≤ θ3
YES
NO
BEST EFFORT CLASS QUEUE LENGTH + PACKET LENGTH ≤ θ2
NO
YES
BAND GUARANTEED CLASS QUEUE LENGTH = BAND GUARANTEED CLASS QUEUE LENGTH + PACKET LENGTH
BEST EFFORT CLASS QUEUE LENGTH = BEST EFFORT CLASS QUEUE LENGTH + PACKET LENGTH
WRITE VARIABLE-LENGTH PACKET INTO BAND GUARANTEED CLASS QUEUE
DROP VARIABLE-LENGTH PACKET
WRITE VARIABLE-LENGTH PACKET INTO BEST EFFORT CLASS QUEUE

FIG. 7

T0020 "922T860

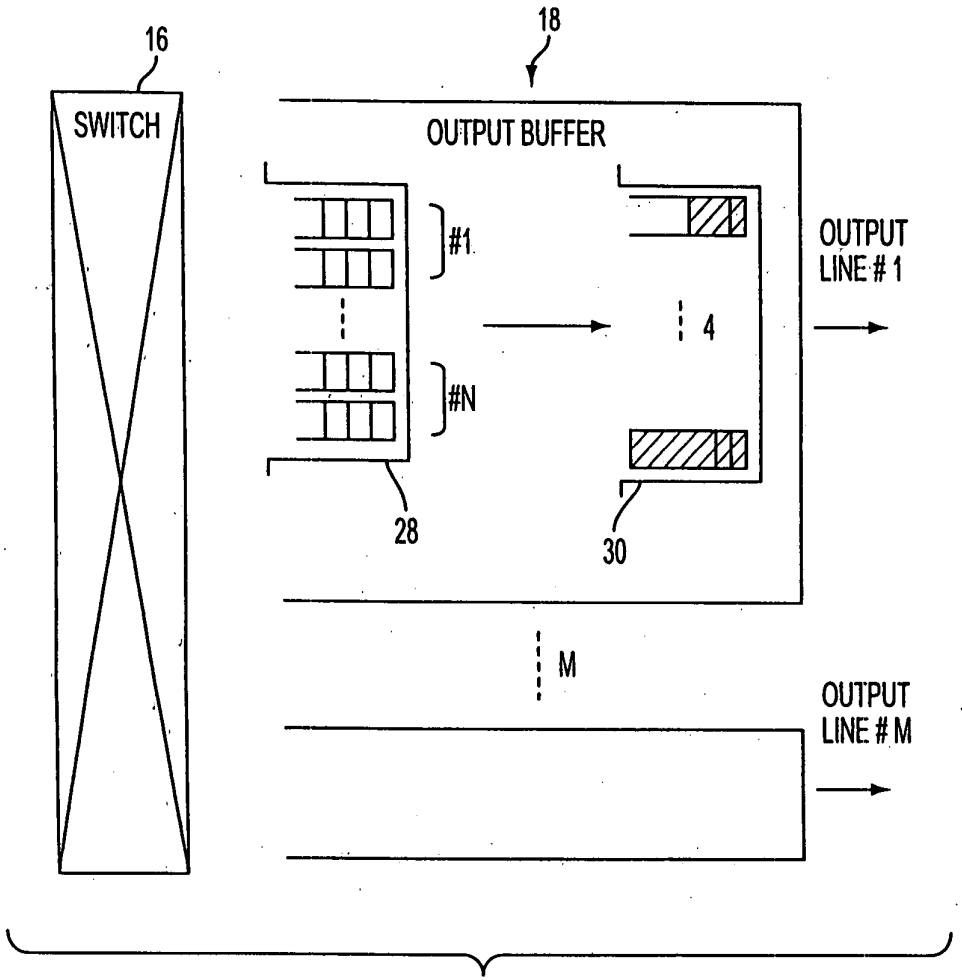


FIG. 8

T0026E0 9226T860

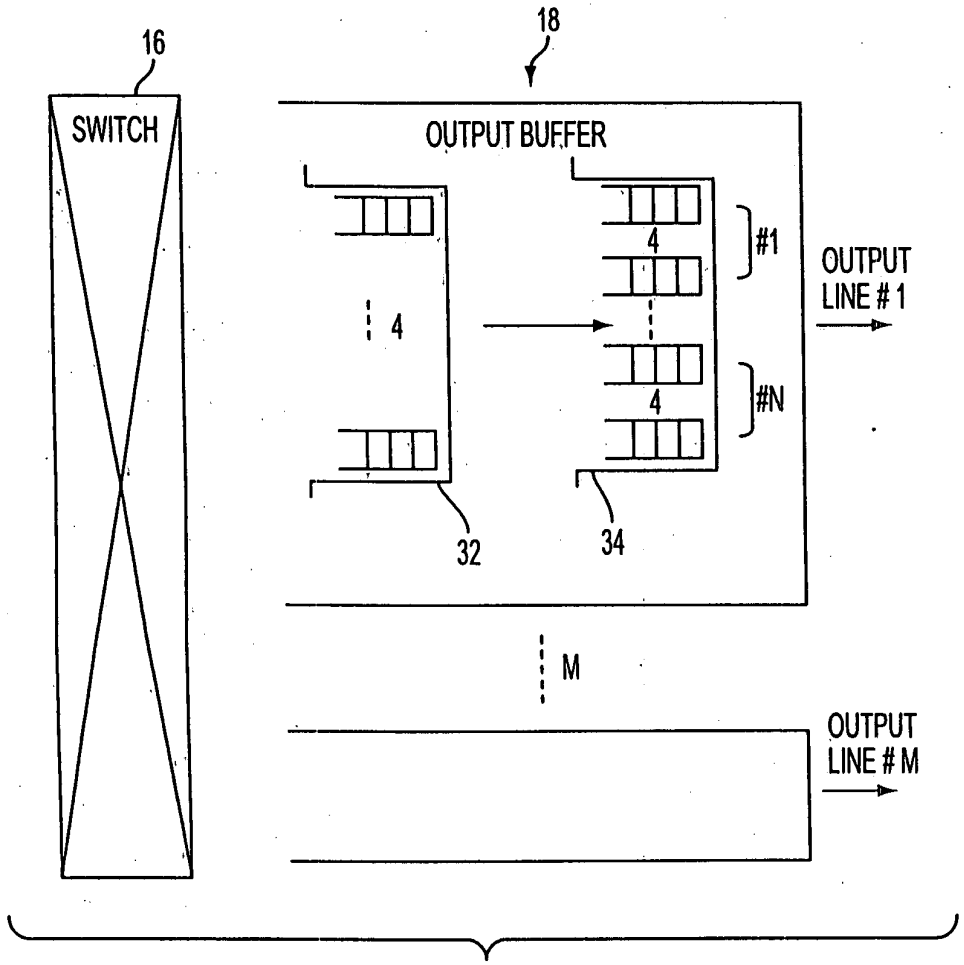


FIG. 9

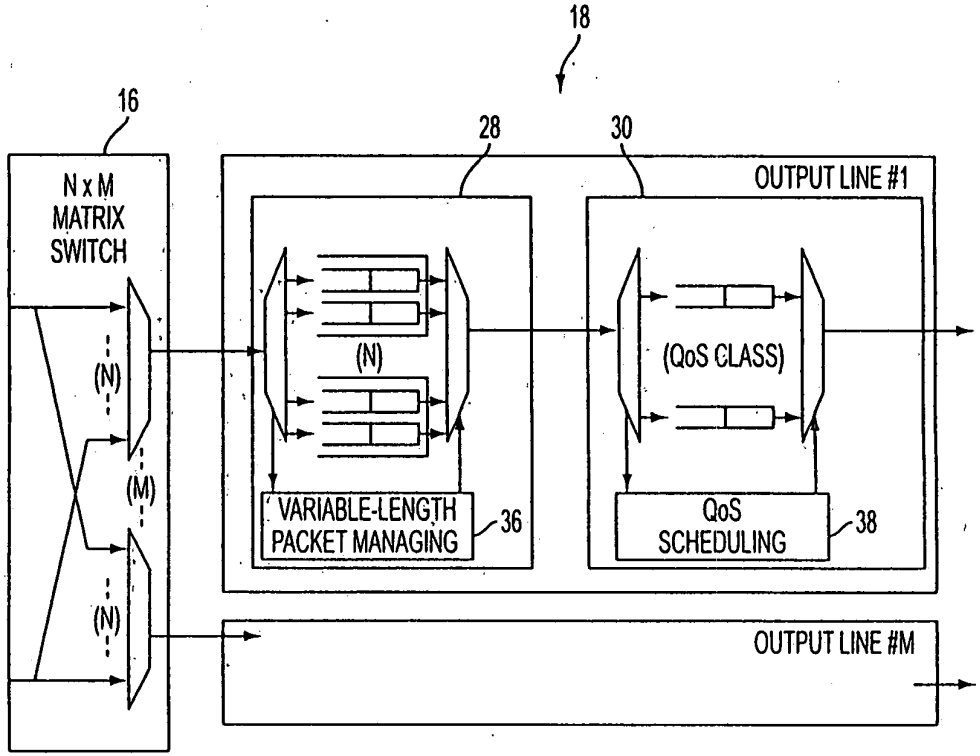

18
16
28
30
N x M
MATRIX
SWITCH
OUTPUT LINE #1
(N)
(M)
(N)
(N)
(QoS CLASS)
VARIABLE-LENGTH
PACKET MANAGING
36
QoS
SCHEDULING
38
OUTPUT LINE #M

FIG. 10

TO020E0 9E2E1860

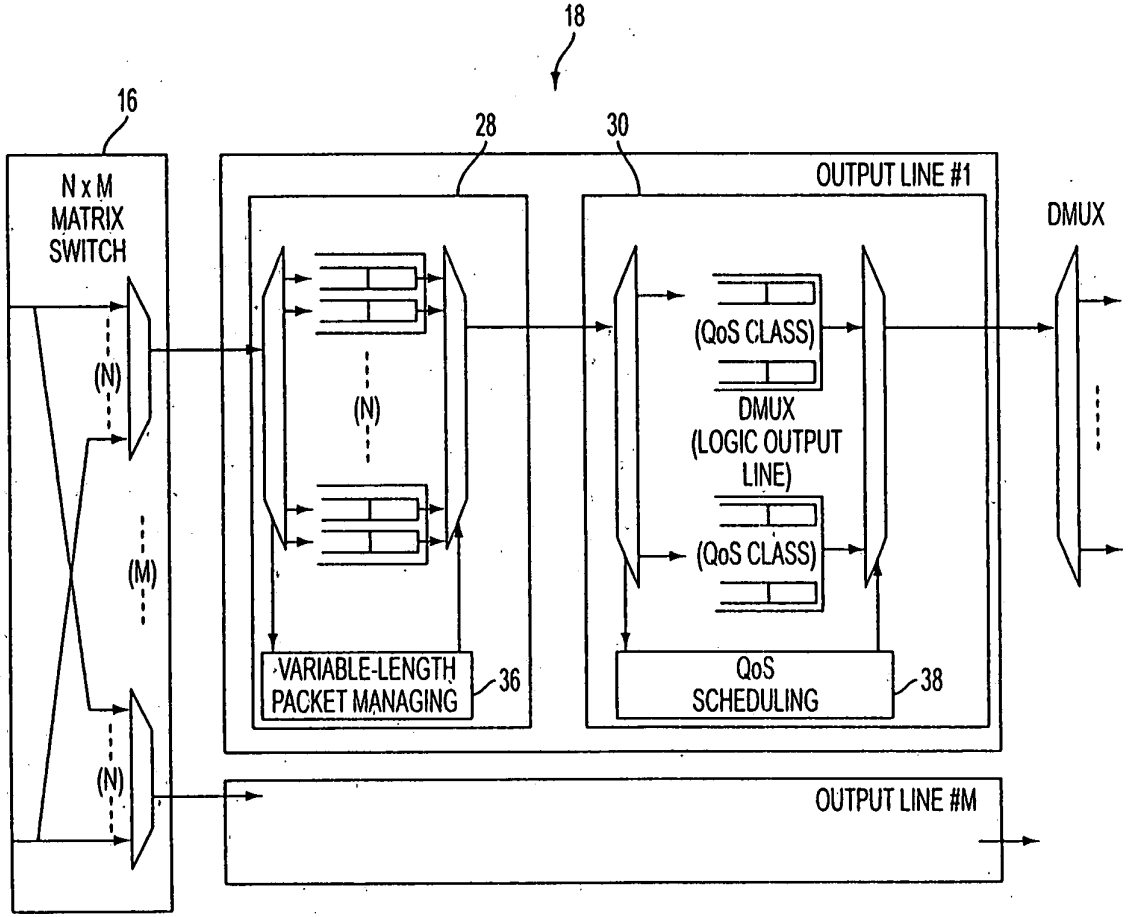


FIG. 11

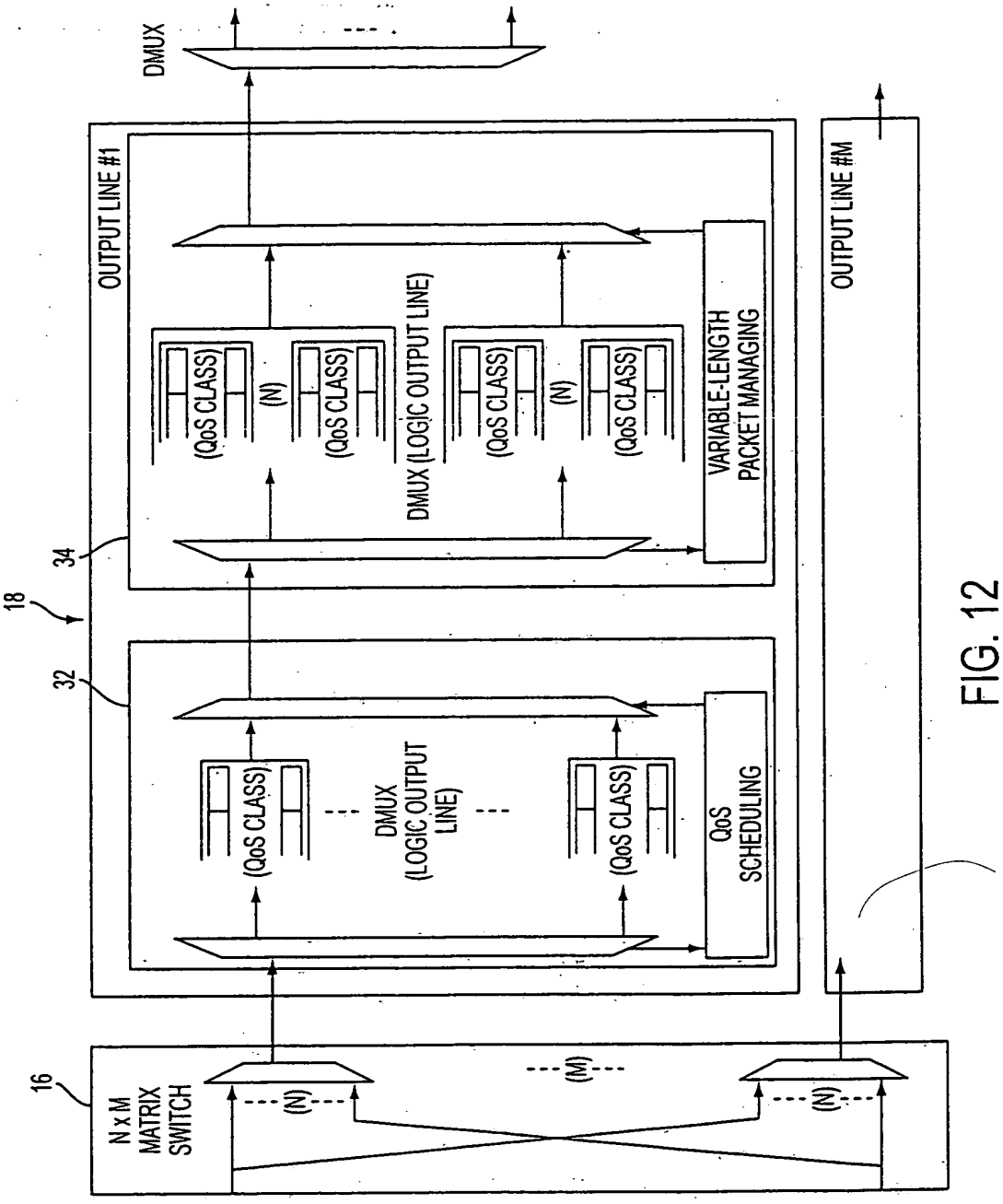

16
N x M
MATRIX
SWITCH
(N)
(M)
(N)
18
OUTPUT LINE #1
OUTPUT LINE #M
32
(QoS CLASS)
DMUX
(LOGIC OUTPUT
LINE)
(QoS CLASS)
QoS
SCHEDULING
34
(QoS CLASS)
(N)
(QoS CLASS)
DMUX (LOGIC OUTPUT LINE)
(QoS CLASS)
(N)
(QoS CLASS)
VARIABLE-LENGTH
PACKET MANAGING
DMUX

FIG. 12

FOREIGN PATENT

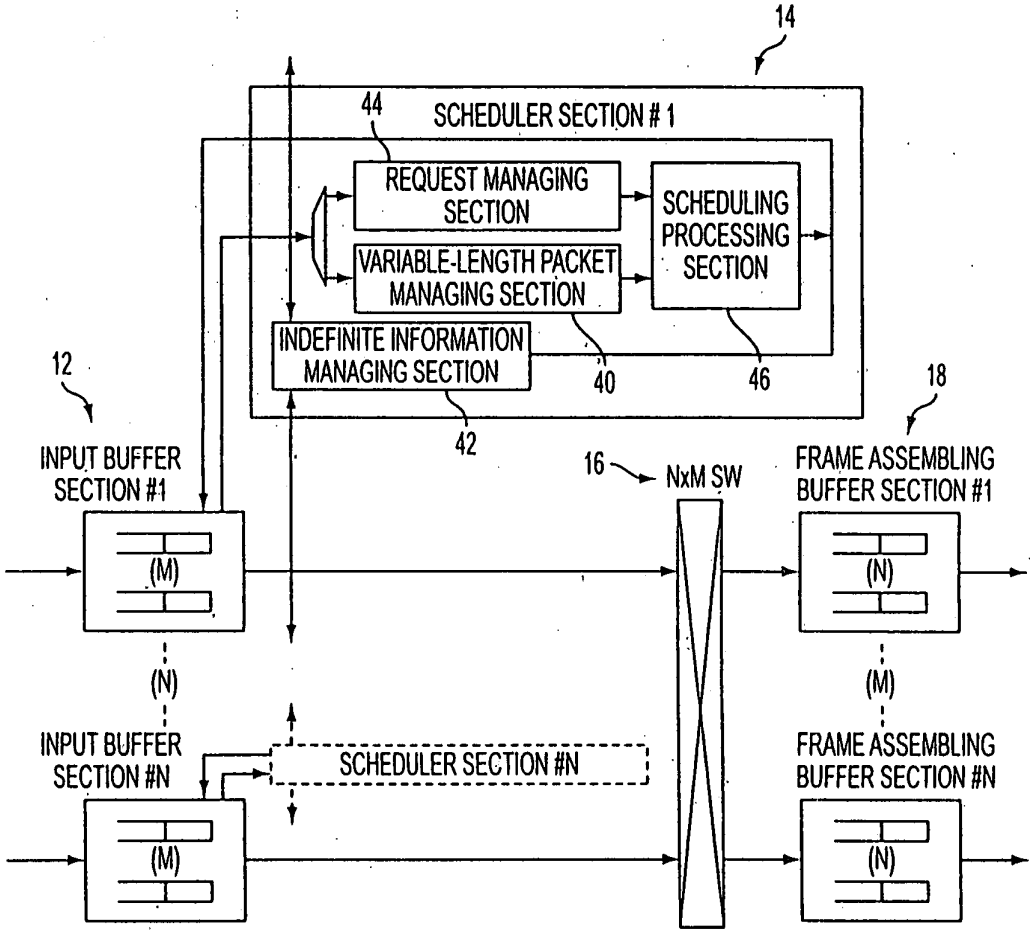


FIG. 13

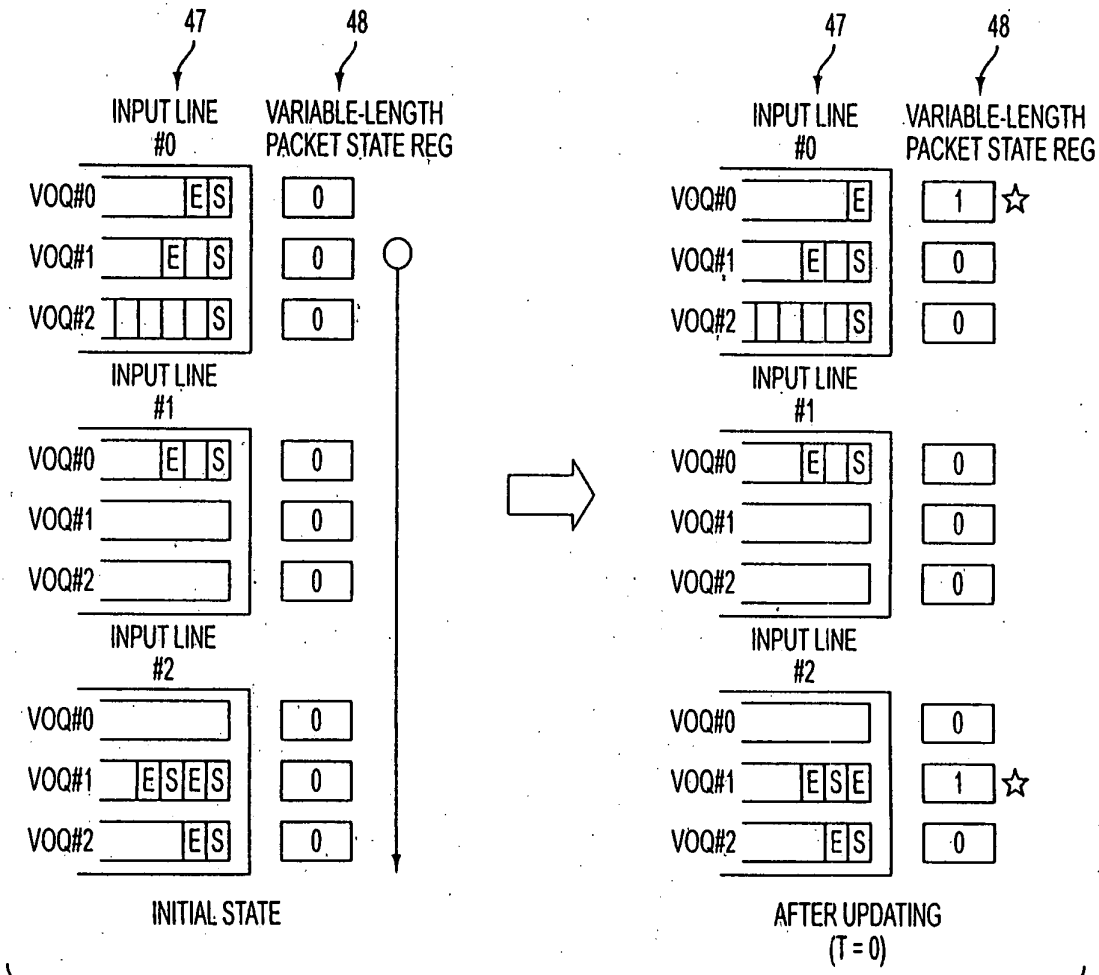

47
48
INPUT LINE #0
VARIABLE-LENGTH PACKET STATE REG
VOQ#0
VOQ#1
VOQ#2
INPUT LINE #1
INPUT LINE #2
INITIAL STATE
AFTER UPDATING (T = 0)

FIG. 14

T0820 92870

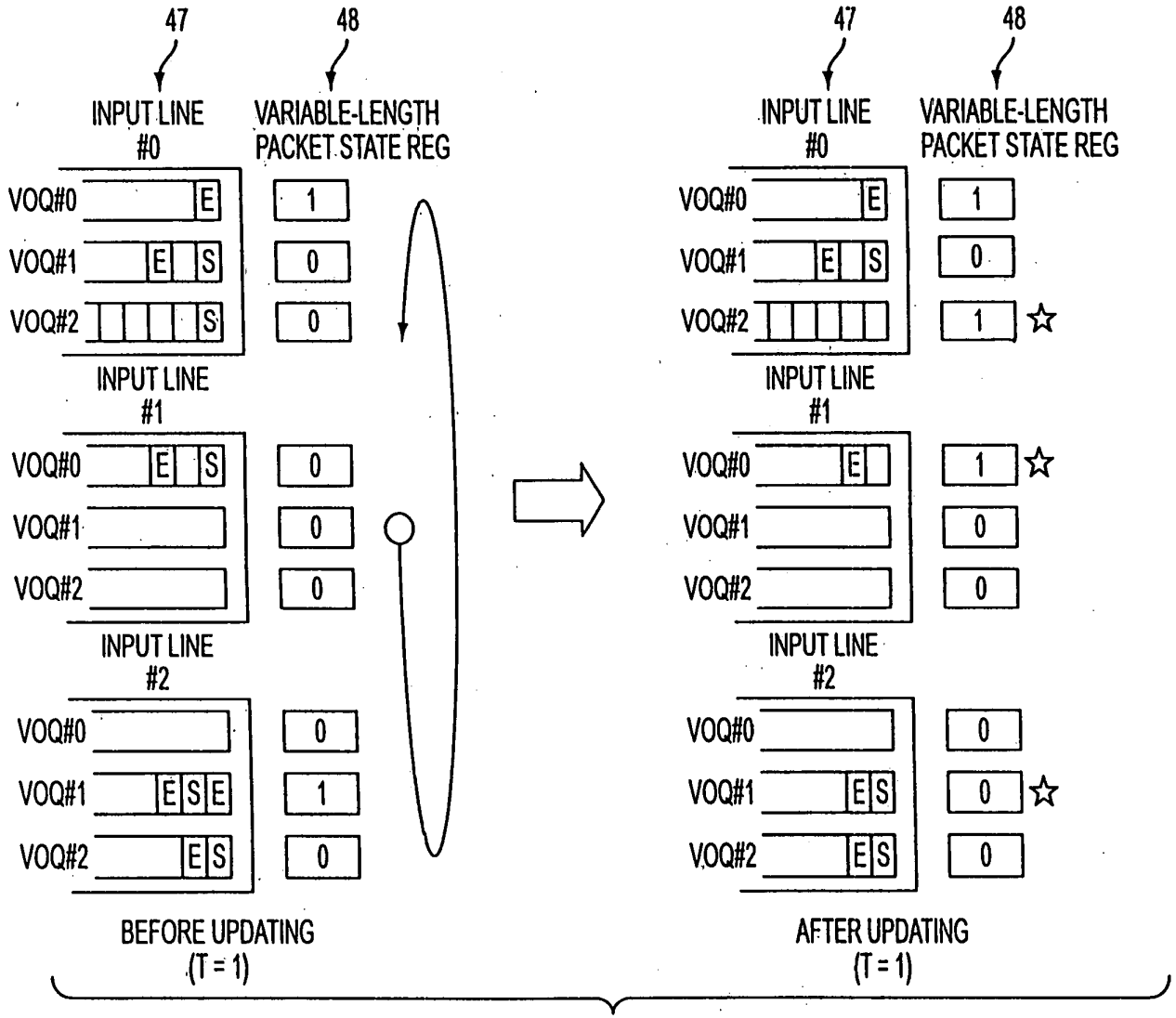


FIG. 15

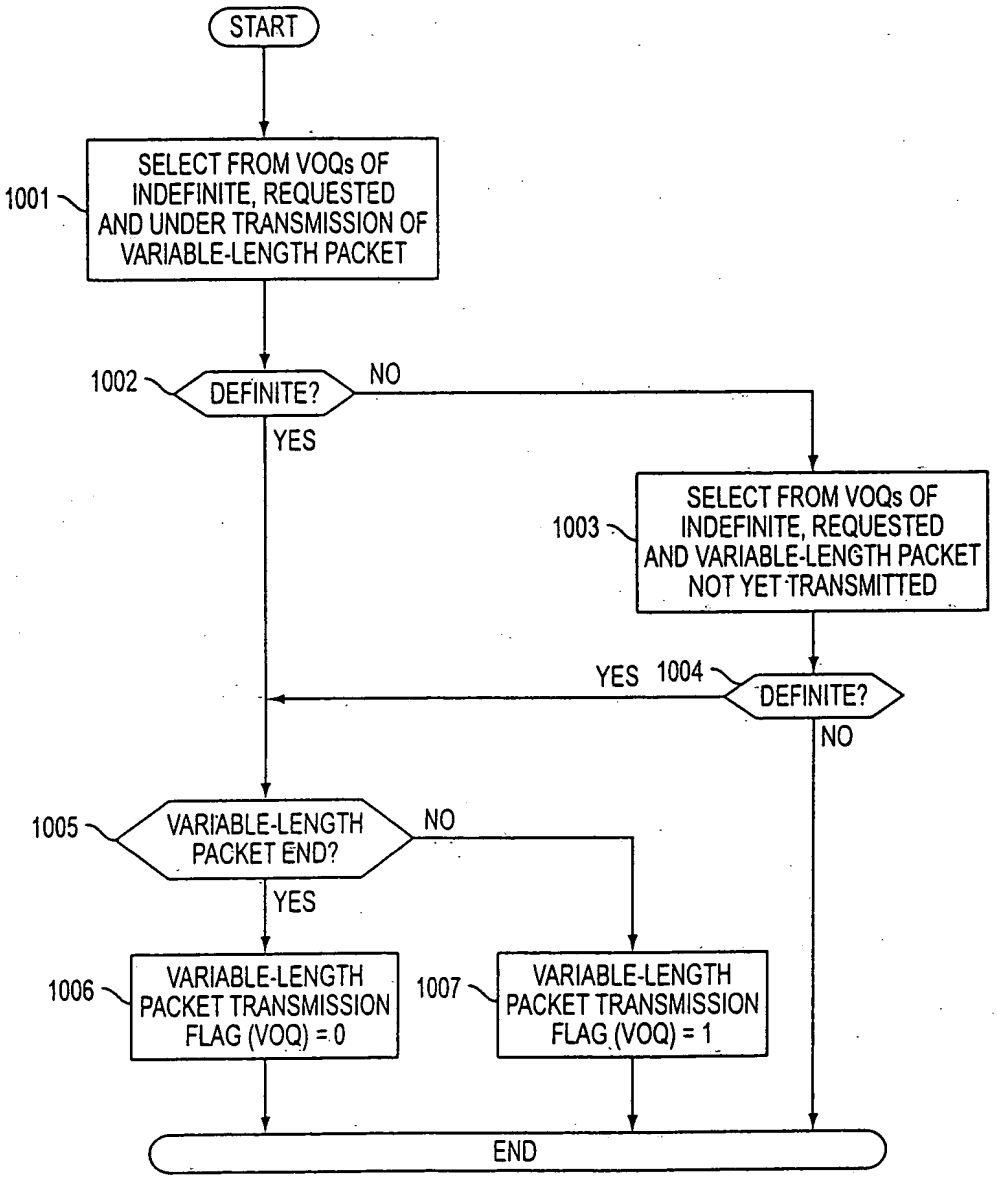

START
SELECT FROM VOQs OF INDEFINITE, REQUESTED AND UNDER TRANSMISSION OF VARIABLE-LENGTH PACKET
1001
1002
DEFINITE?
NO
YES
SELECT FROM VOQs OF INDEFINITE, REQUESTED AND VARIABLE-LENGTH PACKET NOT YET TRANSMITTED
1003
YES
1004
DEFINITE?
NO
1005
VARIABLE-LENGTH PACKET END?
NO
YES
1006
VARIABLE-LENGTH PACKET TRANSMISSION FLAG (VOQ) = 0
1007
VARIABLE-LENGTH PACKET TRANSMISSION FLAG (VOQ) = 1
END

FIG. 16

T0020 9220860

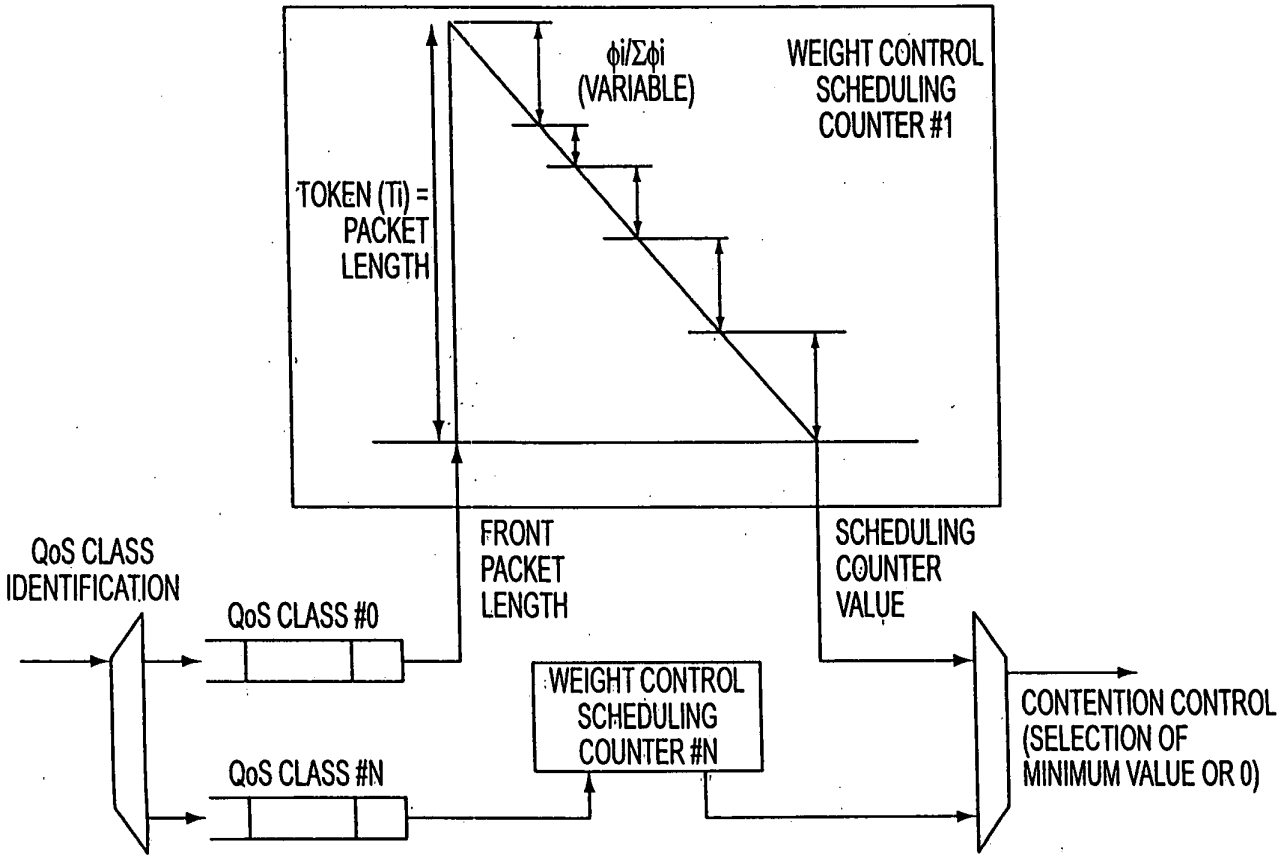


FIG. 17

T0020 922T860

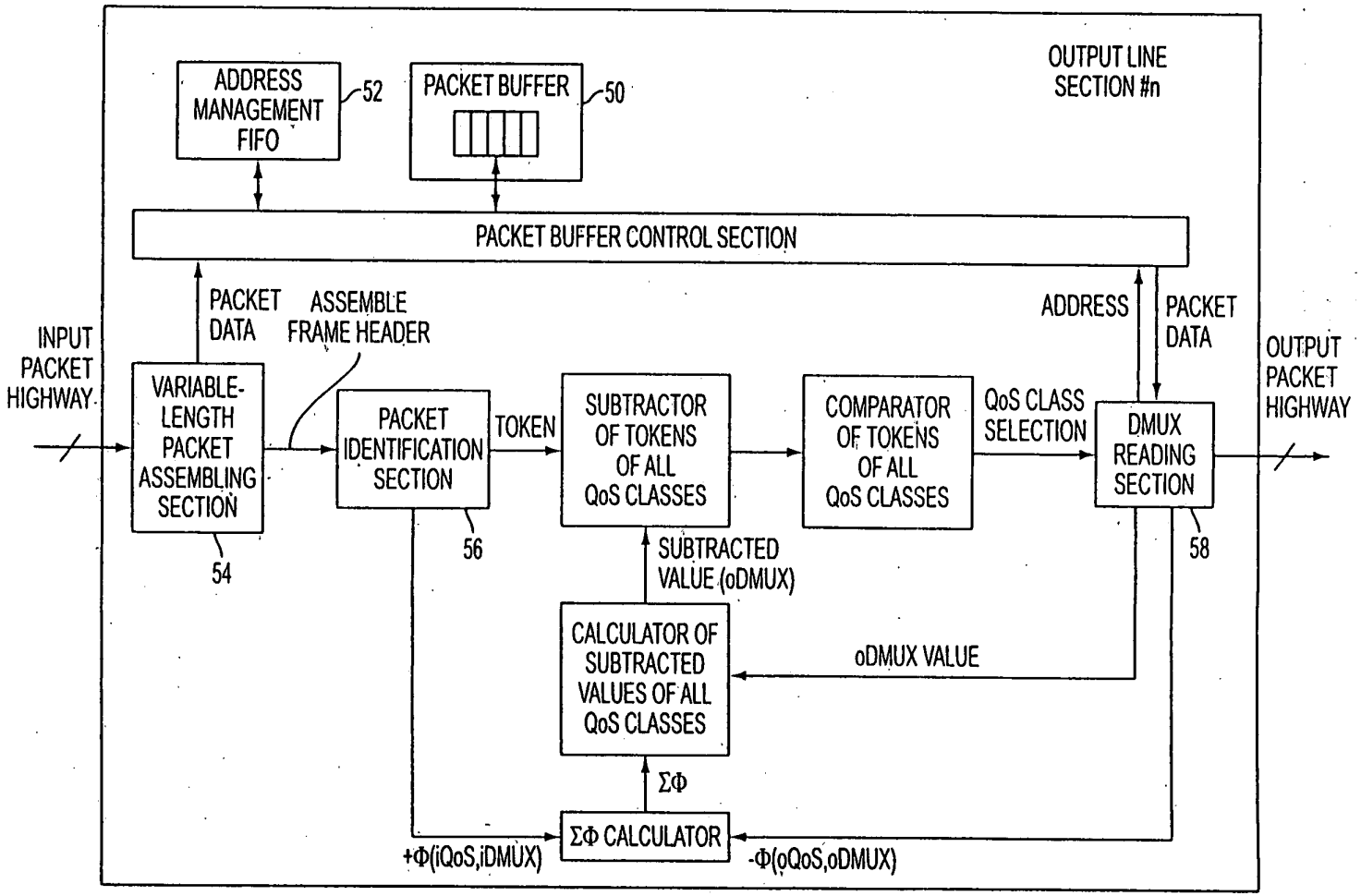


FIG. 18

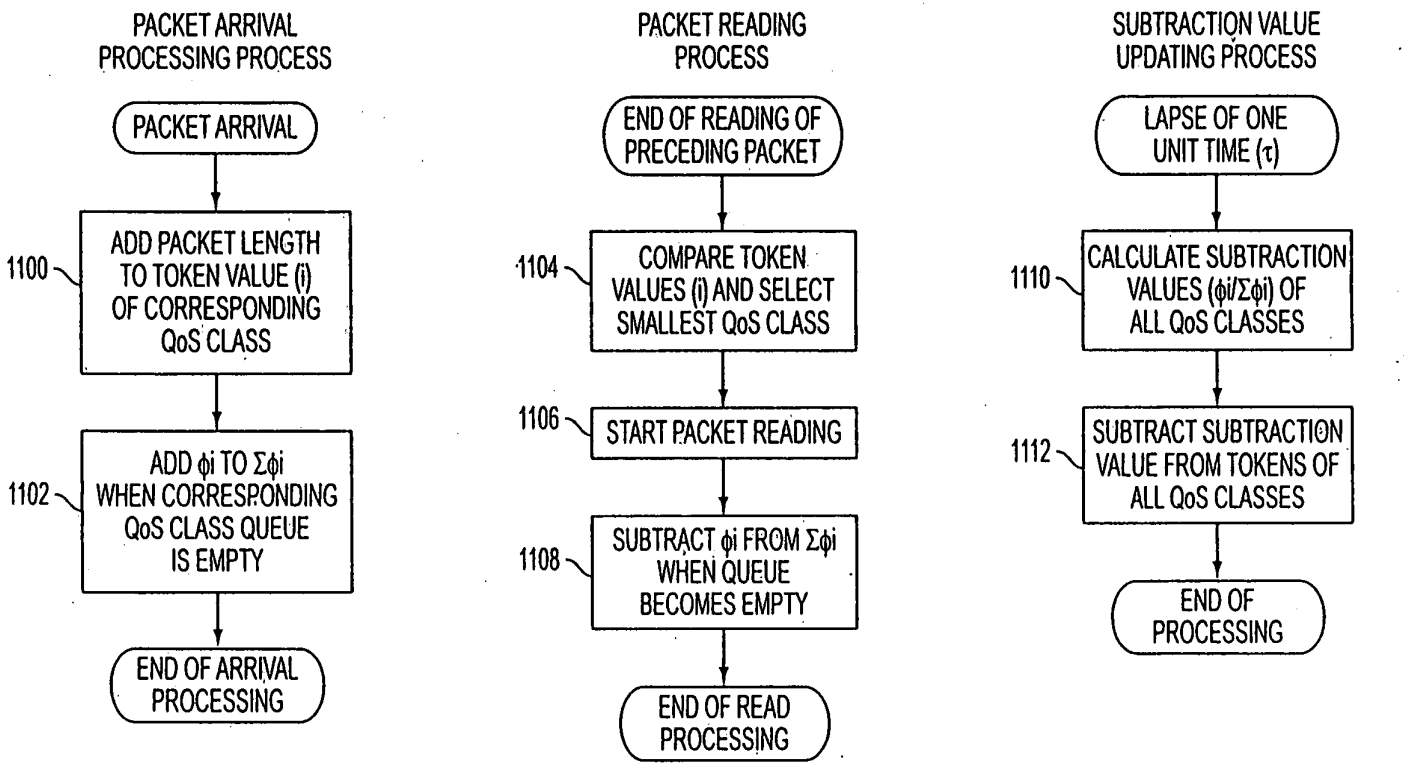

PACKET ARRIVAL PROCESSING PROCESS
PACKET ARRIVAL
1100
ADD PACKET LENGTH TO TOKEN VALUE (i) OF CORRESPONDING QoS CLASS
1102
ADD ϕi TO Σϕi WHEN CORRESPONDING QoS CLASS QUEUE IS EMPTY
END OF ARRIVAL PROCESSING
PACKET READING PROCESS
END OF READING OF PRECEDING PACKET
1104
COMPARE TOKEN VALUES (i) AND SELECT SMALLEST QoS CLASS
1106
START PACKET READING
1108
SUBTRACT ϕi FROM Σϕi WHEN QUEUE BECOMES EMPTY
END OF READ PROCESSING
SUBTRACTION VALUE UPDATING PROCESS
LAPSE OF ONE UNIT TIME (τ)
1110
CALCULATE SUBTRACTION VALUES (ϕi/Σϕi) OF ALL QoS CLASSES
1112
SUBTRACT SUBTRACTION VALUE FROM TOKENS OF ALL QoS CLASSES
END OF PROCESSING

FIG. 19

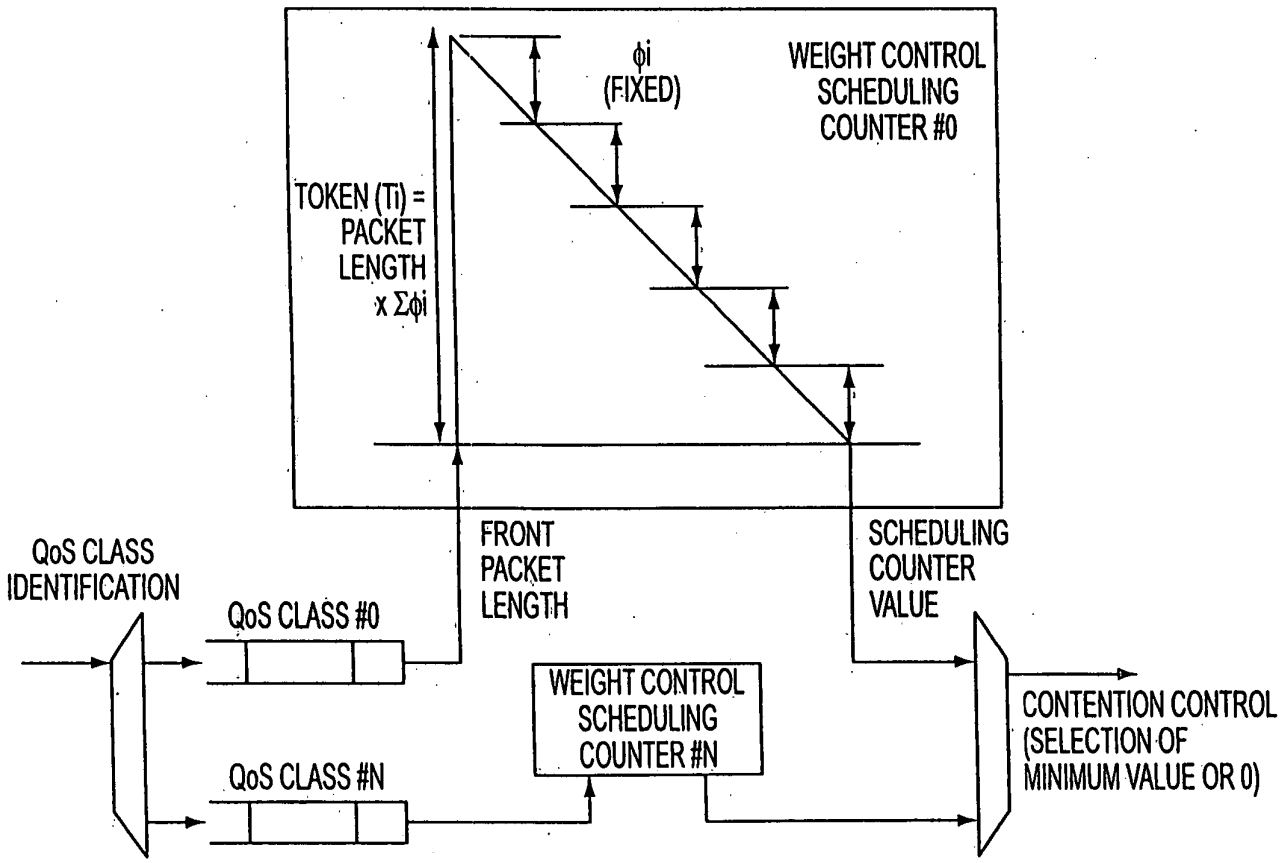

φi
(FIXED)
WEIGHT CONTROL
SCHEDULING
COUNTER #0
TOKEN (Ti) =
PACKET
LENGTH
x Σφi
QoS CLASS
IDENTIFICATION
QoS CLASS #0
FRONT
PACKET
LENGTH
SCHEDULING
COUNTER
VALUE
WEIGHT CONTROL
SCHEDULING
COUNTER #N
CONTENTION CONTROL
(SELECTION OF
MINIMUM VALUE OR 0)
QoS CLASS #N

FIG. 20

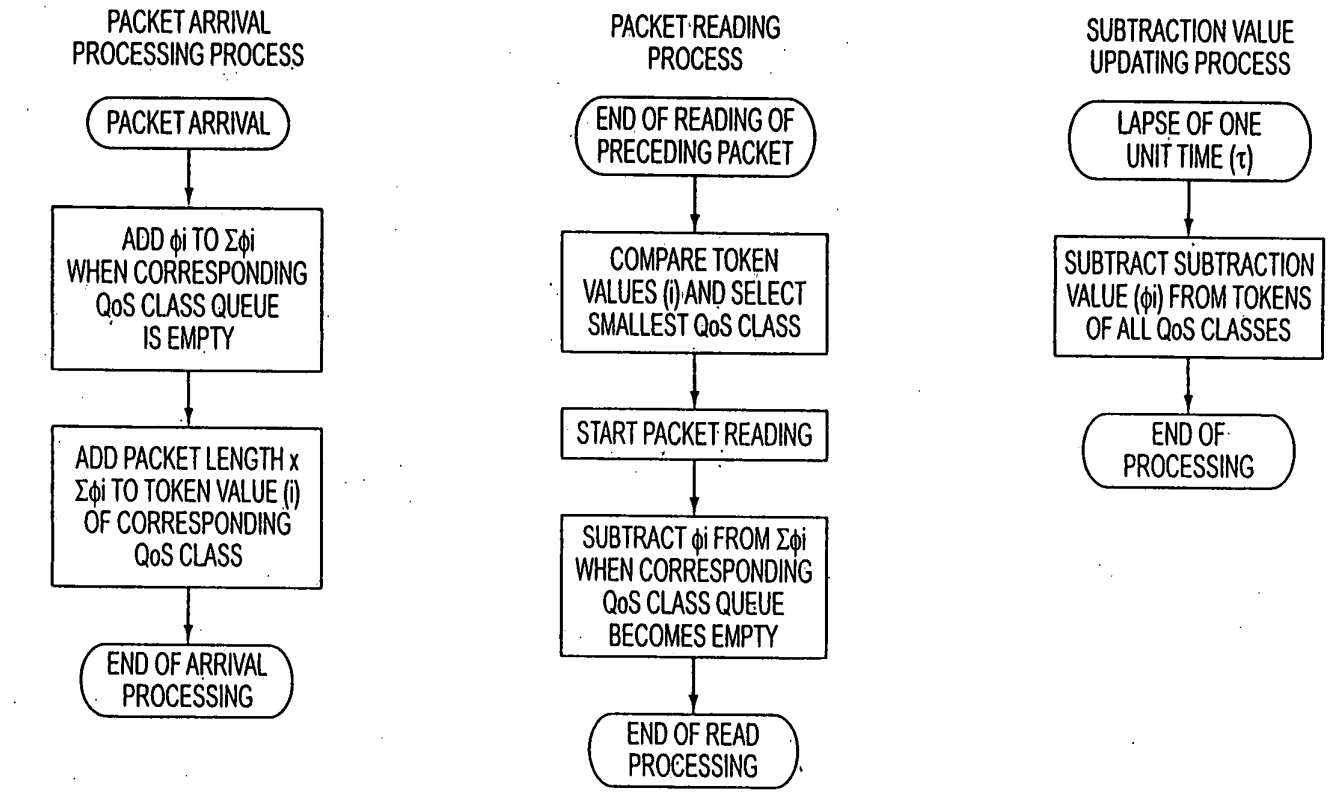

PACKET ARRIVAL PROCESSING PROCESS
PACKET ARRIVAL
ADD φi TO Σφi WHEN CORRESPONDING QoS CLASS QUEUE IS EMPTY
ADD PACKET LENGTH x Σφi TO TOKEN VALUE (i) OF CORRESPONDING QoS CLASS
END OF ARRIVAL PROCESSING
PACKET READING PROCESS
END OF READING OF PRECEDING PACKET
COMPARE TOKEN VALUES (i) AND SELECT SMALLEST QoS CLASS
START PACKET READING
SUBTRACT φi FROM Σφi WHEN CORRESPONDING QoS CLASS QUEUE BECOMES EMPTY
END OF READ PROCESSING
SUBTRACTION VALUE UPDATING PROCESS
LAPSE OF ONE UNIT TIME (τ)
SUBTRACT SUBTRACTION VALUE (φi) FROM TOKENS OF ALL QoS CLASSES
END OF PROCESSING

FIG. 21

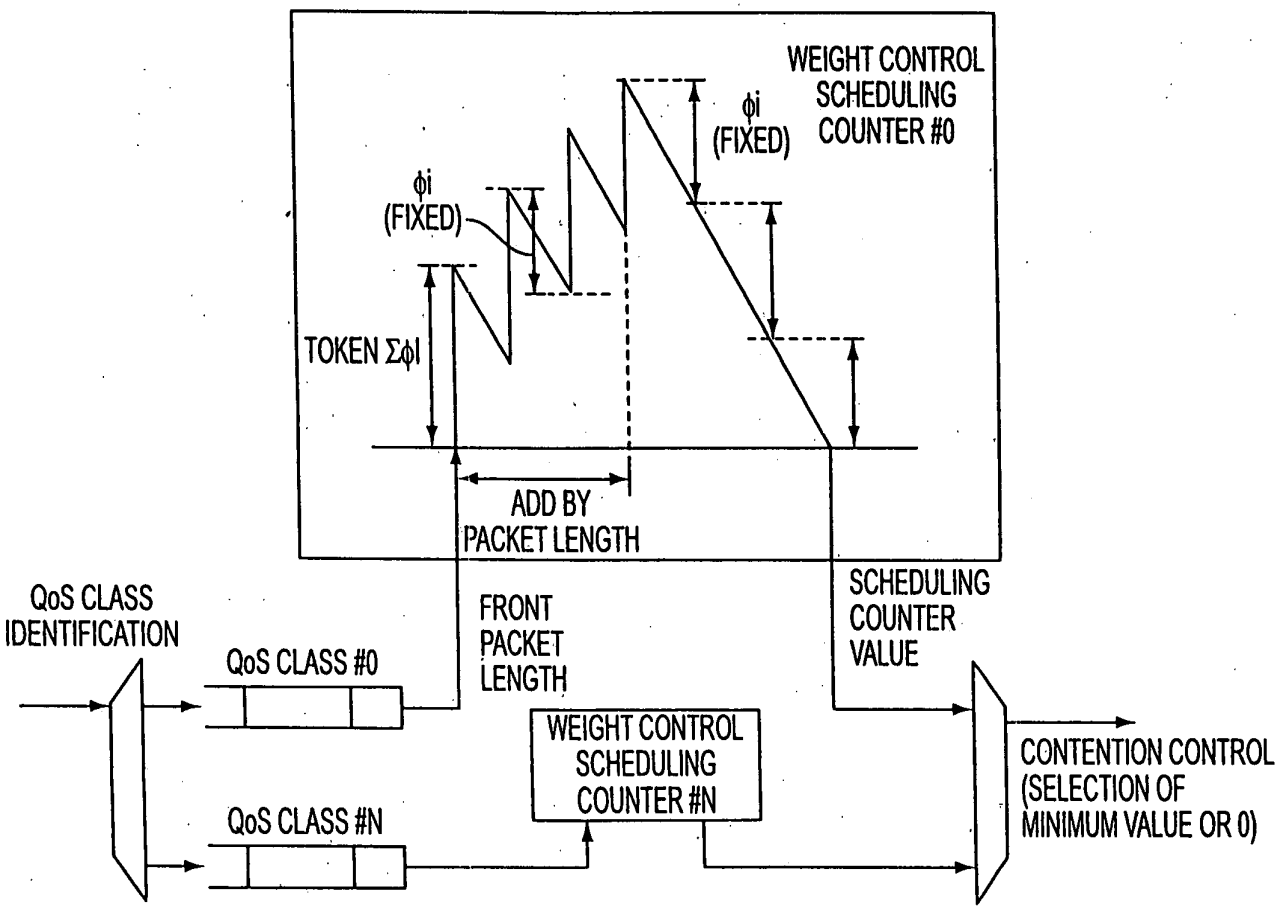

WEIGHT CONTROL SCHEDULING COUNTER #0
φi (FIXED)
φi (FIXED)
TOKEN Σφi
ADD BY PACKET LENGTH
QoS CLASS IDENTIFICATION
QoS CLASS #0
FRONT PACKET LENGTH
SCHEDULING COUNTER VALUE
WEIGHT CONTROL SCHEDULING COUNTER #N
QoS CLASS #N
CONTENTION CONTROL (SELECTION OF MINIMUM VALUE OR 0)

FIG. 22

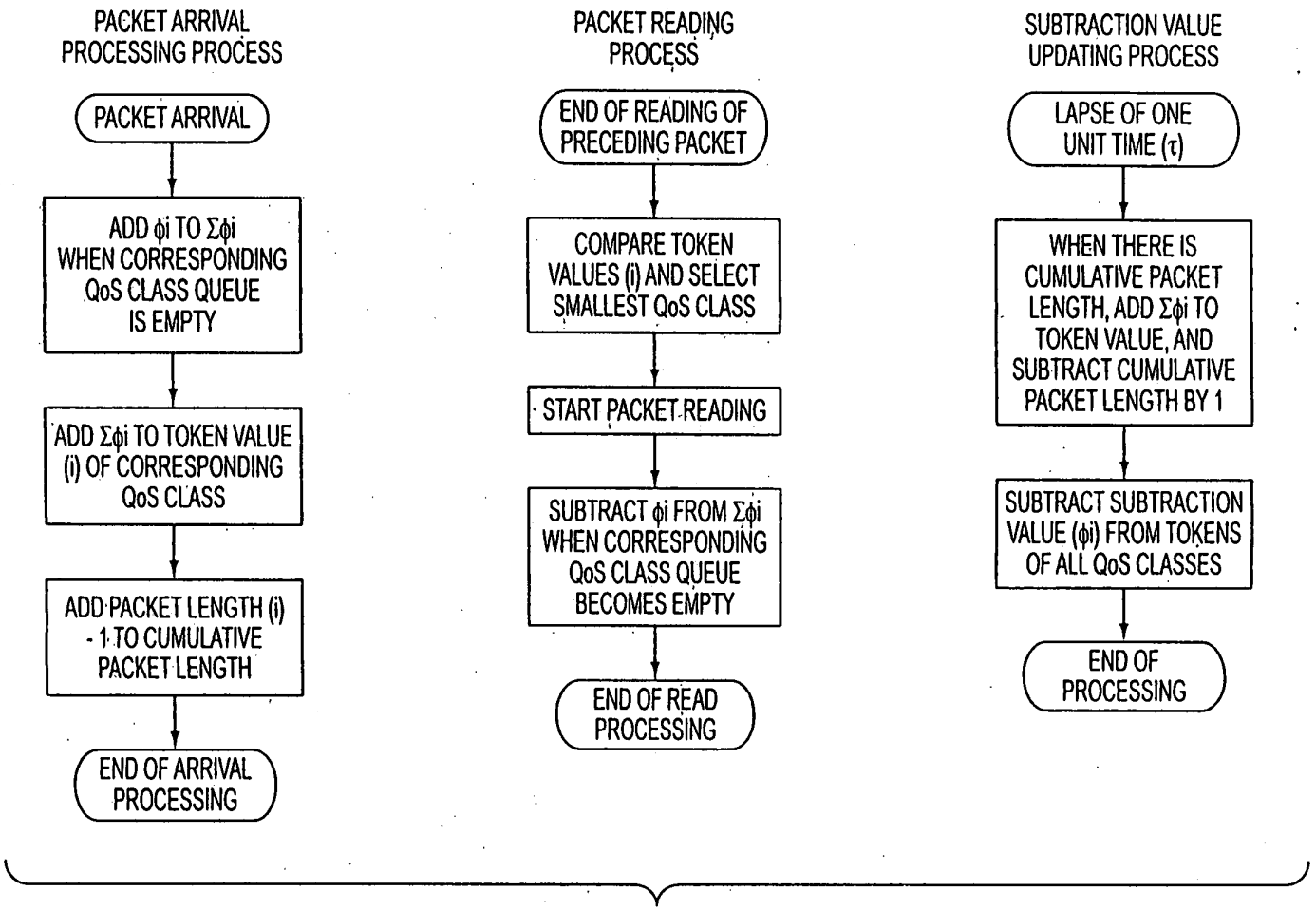

PACKET ARRIVAL PROCESSING PROCESS
PACKET ARRIVAL
ADD ϕi TO Σϕi WHEN CORRESPONDING QoS CLASS QUEUE IS EMPTY
ADD Σϕi TO TOKEN VALUE (i) OF CORRESPONDING QoS CLASS
ADD PACKET LENGTH (i) - 1 TO CUMULATIVE PACKET LENGTH
END OF ARRIVAL PROCESSING
PACKET READING PROCESS
END OF READING OF PRECEDING PACKET
COMPARE TOKEN VALUES (i) AND SELECT SMALLEST QoS CLASS
START PACKET READING
SUBTRACT ϕi FROM Σϕi WHEN CORRESPONDING QoS CLASS QUEUE BECOMES EMPTY
END OF READ PROCESSING
SUBTRACTION VALUE UPDATING PROCESS
LAPSE OF ONE UNIT TIME (τ)
WHEN THERE IS CUMULATIVE PACKET LENGTH, ADD Σϕi TO TOKEN VALUE, AND SUBTRACT CUMULATIVE PACKET LENGTH BY 1
SUBTRACT SUBTRACTION VALUE (ϕi) FROM TOKENS OF ALL QoS CLASSES
END OF PROCESSING

FIG. 23

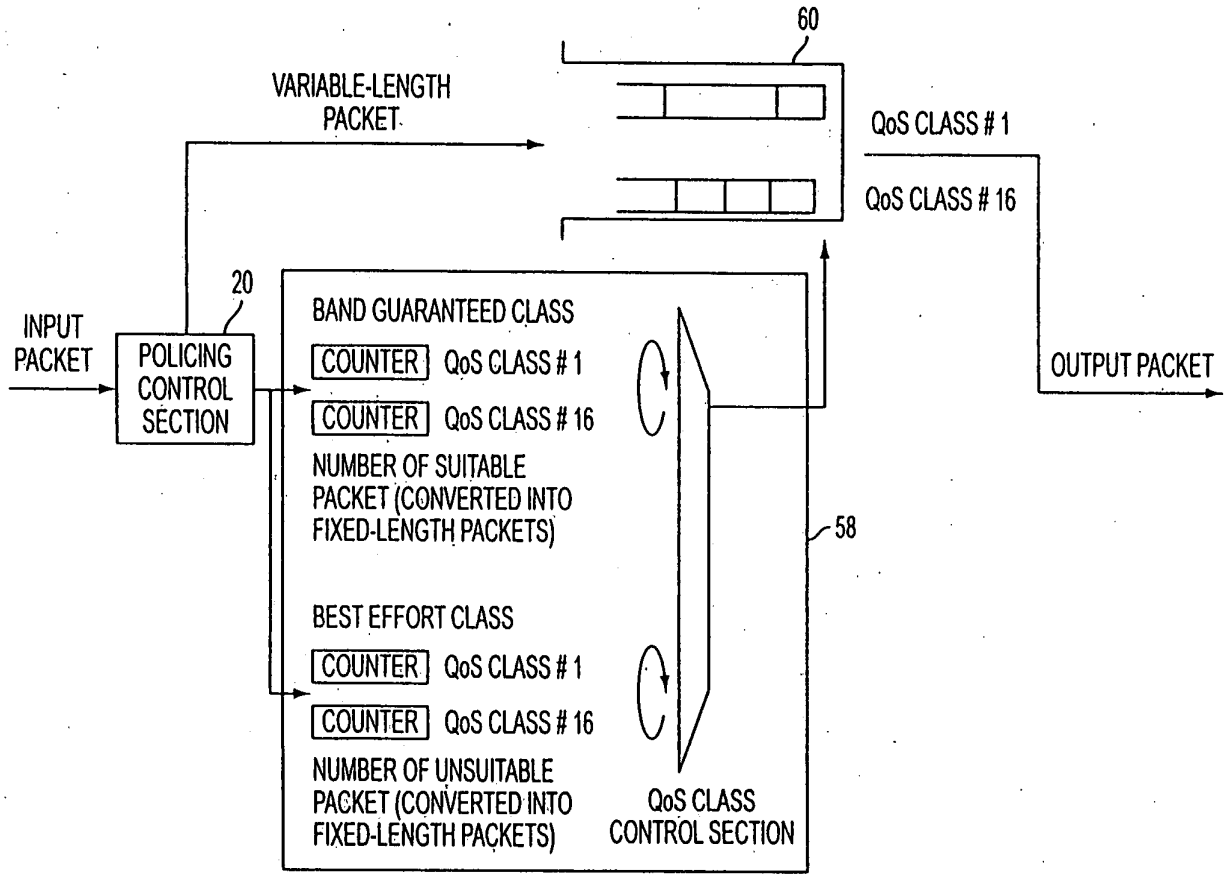

60
VARIABLE-LENGTH PACKET
QoS CLASS # 1
QoS CLASS # 16
20
INPUT PACKET
POLICING CONTROL SECTION
BAND GUARANTEED CLASS
COUNTER QoS CLASS # 1
COUNTER QoS CLASS # 16
NUMBER OF SUITABLE PACKET (CONVERTED INTO FIXED-LENGTH PACKETS)
BEST EFFORT CLASS
COUNTER QoS CLASS # 1
COUNTER QoS CLASS # 16
NUMBER OF UNSUITABLE PACKET (CONVERTED INTO FIXED-LENGTH PACKETS)
58
QoS CLASS CONTROL SECTION
OUTPUT PACKET

FIG. 24

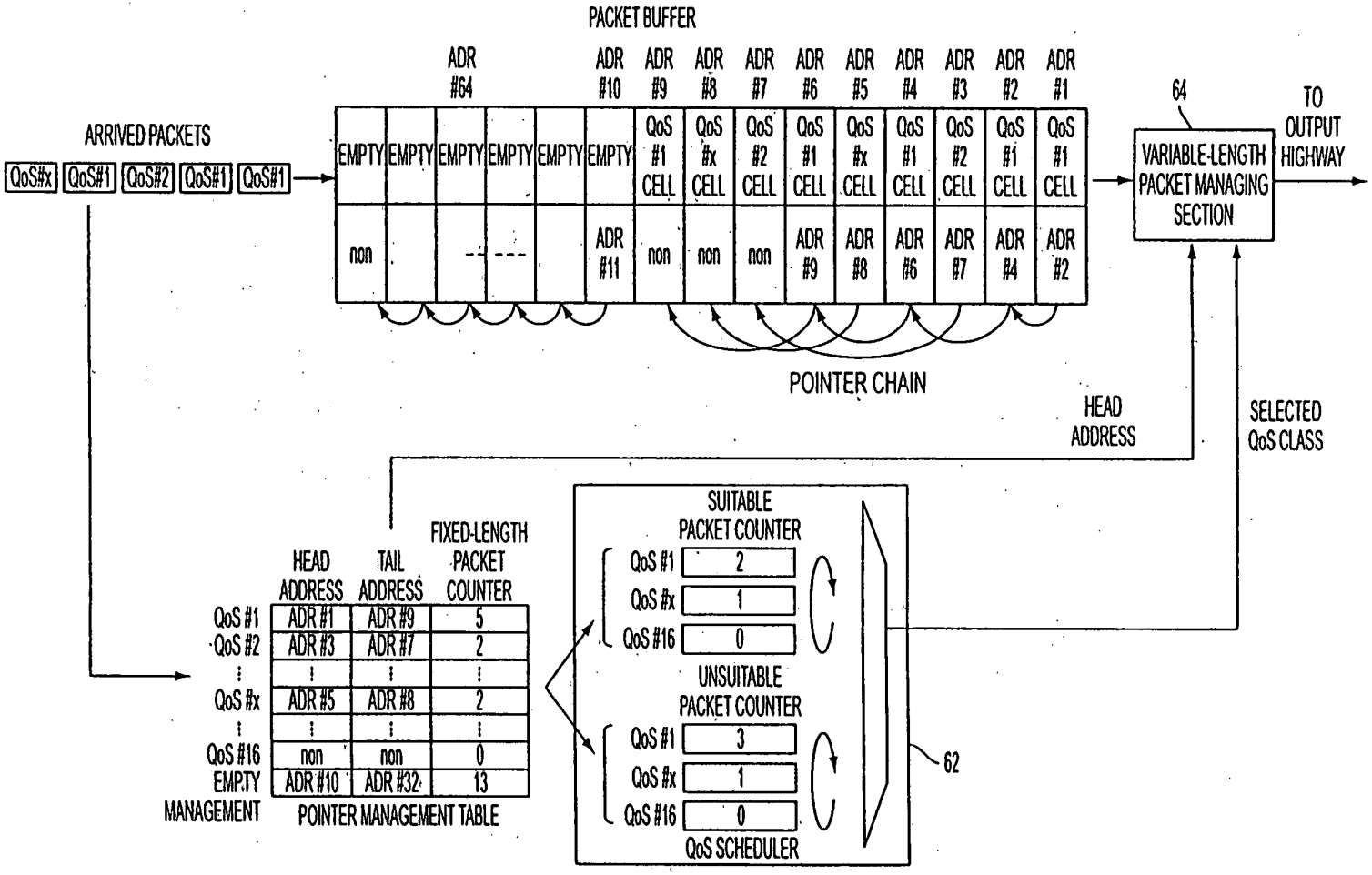

PACKET BUFFER
ADR #64
ADR #10
ADR #9
ADR #8
ADR #7
ADR #6
ADR #5
ADR #4
ADR #3
ADR #2
ADR #1
ARRIVED PACKETS
QoS#x
QoS#1
QoS#2
QoS#1
QoS#1
EMPTY
QoS #1 CELL
QoS #x CELL
QoS #2 CELL
non
ADR #11
ADR #9
ADR #8
ADR #6
ADR #7
ADR #4
ADR #2
64
VARIABLE-LENGTH PACKET MANAGING SECTION
TO OUTPUT HIGHWAY
POINTER CHAIN
HEAD ADDRESS
SELECTED QoS CLASS
FIXED-LENGTH PACKET COUNTER
HEAD ADDRESS
TAIL ADDRESS
QoS #1
QoS #2
QoS #x
QoS #16
EMPTY MANAGEMENT
ADR #1
ADR #9
5
ADR #3
ADR #7
2
ADR #5
ADR #8
2
non
0
ADR #10
ADR #32
13
POINTER MANAGEMENT TABLE
SUITABLE PACKET COUNTER
UNSUITABLE PACKET COUNTER
QoS SCHEDULER
62

FIG. 25

FOR2D' 92ETA6D

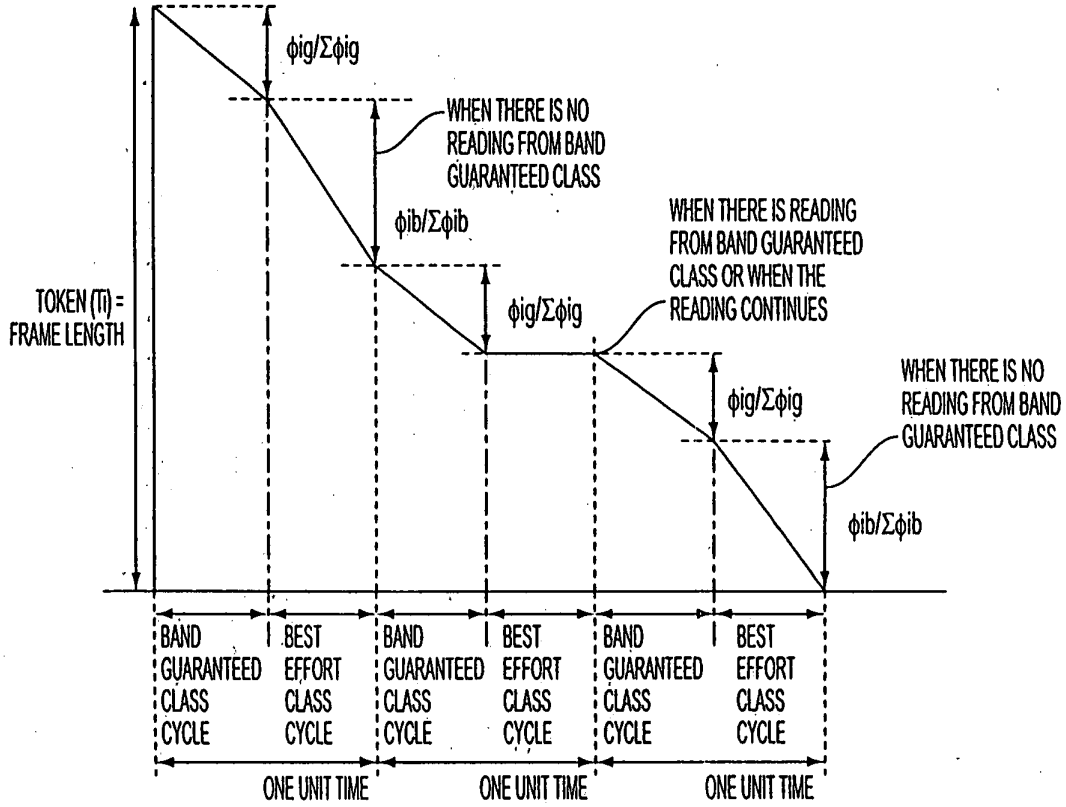


FIG. 26

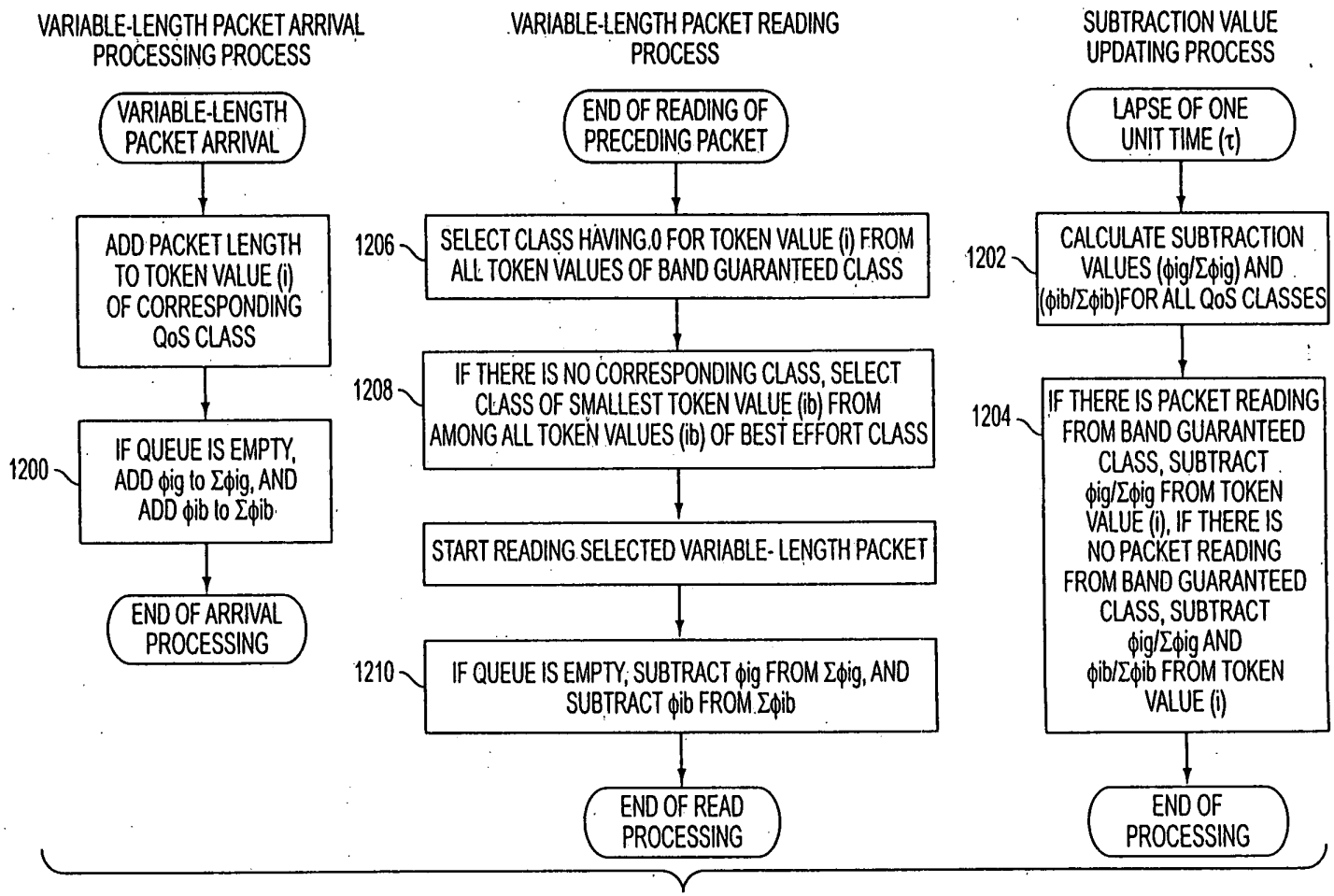

VARIABLE-LENGTH PACKET ARRIVAL PROCESSING PROCESS
VARIABLE-LENGTH PACKET ARRIVAL
ADD PACKET LENGTH TO TOKEN VALUE (i) OF CORRESPONDING QoS CLASS
1200
IF QUEUE IS EMPTY, ADD ϕig to Σϕig, AND ADD ϕib to Σϕib
END OF ARRIVAL PROCESSING
VARIABLE-LENGTH PACKET READING PROCESS
END OF READING OF PRECEDING PACKET
1206
SELECT CLASS HAVING 0 FOR TOKEN VALUE (i) FROM ALL TOKEN VALUES OF BAND GUARANTEED CLASS
1208
IF THERE IS NO CORRESPONDING CLASS, SELECT CLASS OF SMALLEST TOKEN VALUE (ib) FROM AMONG ALL TOKEN VALUES (ib) OF BEST EFFORT CLASS
START READING SELECTED VARIABLE- LENGTH PACKET
1210
IF QUEUE IS EMPTY, SUBTRACT ϕig FROM Σϕig, AND SUBTRACT ϕib FROM Σϕib
END OF READ PROCESSING
SUBTRACTION VALUE UPDATING PROCESS
LAPSE OF ONE UNIT TIME (τ)
1202
CALCULATE SUBTRACTION VALUES (ϕig/Σϕig) AND (ϕib/Σϕib)FOR ALL QoS CLASSES
1204
IF THERE IS PACKET READING FROM BAND GUARANTEED CLASS, SUBTRACT ϕig/Σϕig FROM TOKEN VALUE (i), IF THERE IS NO PACKET READING FROM BAND GUARANTEED CLASS, SUBTRACT ϕig/Σϕig AND ϕib/Σϕib FROM TOKEN VALUE (i)
END OF PROCESSING

FIG. 27

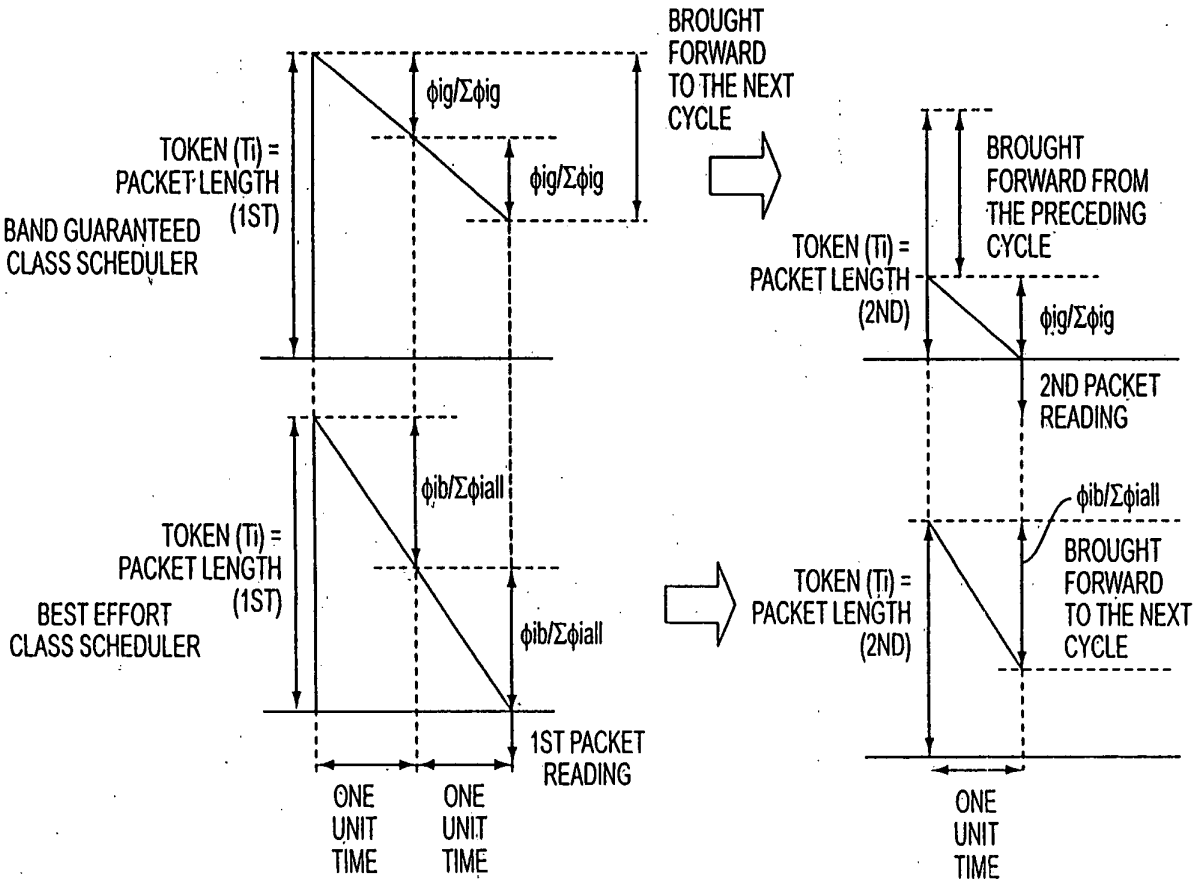

BROUGHT FORWARD TO THE NEXT CYCLE
TOKEN (Ti) = PACKET LENGTH (1ST)
BAND GUARANTEED CLASS SCHEDULER
φig/Σφig
φig/Σφig
BROUGHT FORWARD FROM THE PRECEDING CYCLE
TOKEN (Ti) = PACKET LENGTH (2ND)
φig/Σφig
2ND PACKET READING
φib/Σφiall
TOKEN (Ti) = PACKET LENGTH (1ST)
BEST EFFORT CLASS SCHEDULER
φib/Σφiall
φib/Σφiall
BROUGHT FORWARD TO THE NEXT CYCLE
TOKEN (Ti) = PACKET LENGTH (2ND)
1ST PACKET READING
ONE UNIT TIME
ONE UNIT TIME
ONE UNIT TIME

FIG. 28

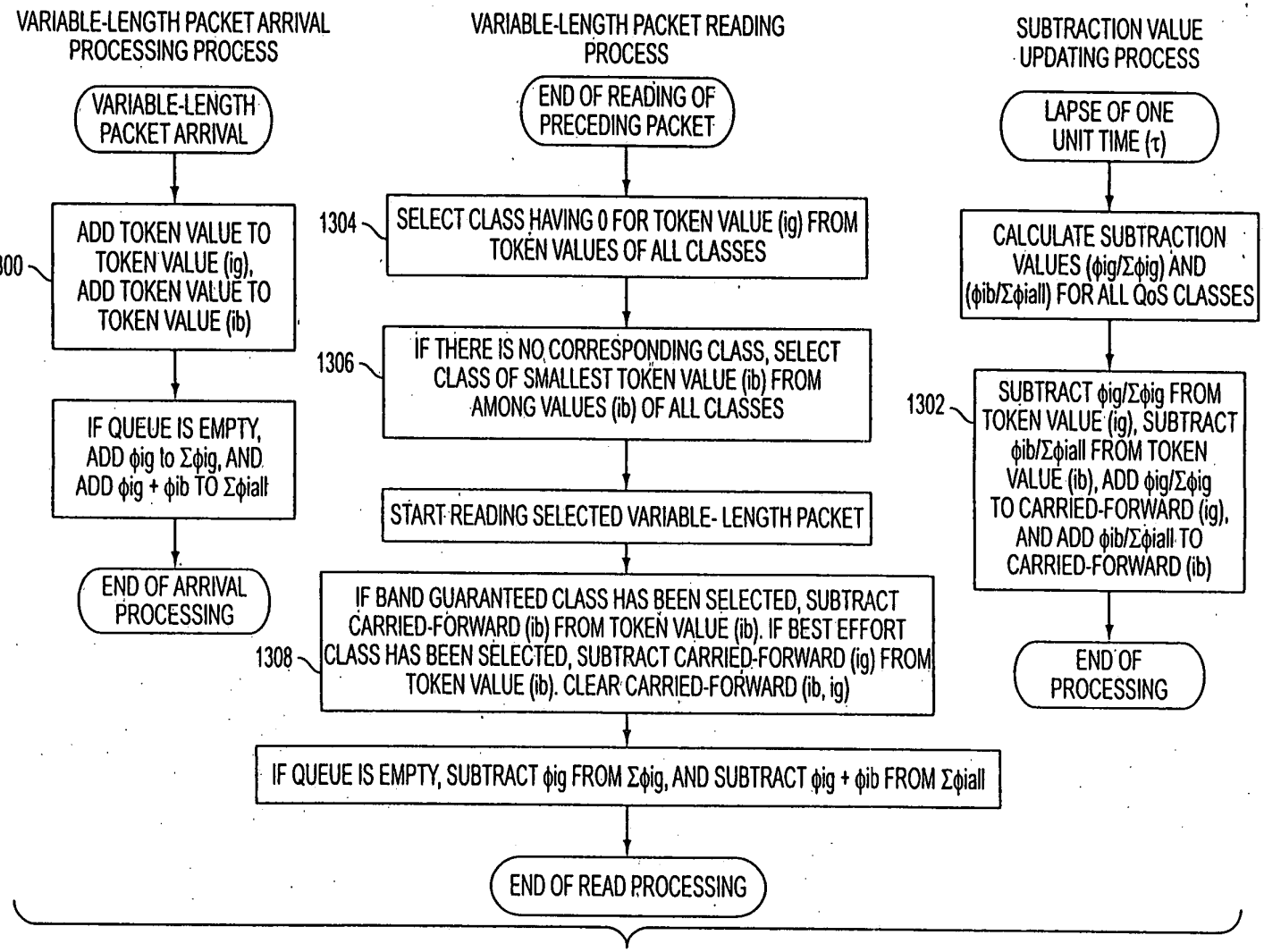

VARIABLE-LENGTH PACKET ARRIVAL PROCESSING PROCESS
VARIABLE-LENGTH PACKET ARRIVAL
ADD TOKEN VALUE TO TOKEN VALUE (ig), ADD TOKEN VALUE TO TOKEN VALUE (ib)
1300
IF QUEUE IS EMPTY, ADD ϕig to Σϕig, AND ADD ϕig + ϕib TO Σϕiall
END OF ARRIVAL PROCESSING
VARIABLE-LENGTH PACKET READING PROCESS
END OF READING OF PRECEDING PACKET
1304
SELECT CLASS HAVING 0 FOR TOKEN VALUE (ig) FROM TOKEN VALUES OF ALL CLASSES
1306
IF THERE IS NO CORRESPONDING CLASS, SELECT CLASS OF SMALLEST TOKEN VALUE (ib) FROM AMONG VALUES (ib) OF ALL CLASSES
START READING SELECTED VARIABLE- LENGTH PACKET
1308
IF BAND GUARANTEED CLASS HAS BEEN SELECTED, SUBTRACT CARRIED-FORWARD (ib) FROM TOKEN VALUE (ib). IF BEST EFFORT CLASS HAS BEEN SELECTED, SUBTRACT CARRIED-FORWARD (ig) FROM TOKEN VALUE (ib). CLEAR CARRIED-FORWARD (ib, ig)
IF QUEUE IS EMPTY, SUBTRACT ϕig FROM Σϕig, AND SUBTRACT ϕig + ϕib FROM Σϕiall
END OF READ PROCESSING
SUBTRACTION VALUE UPDATING PROCESS
LAPSE OF ONE UNIT TIME (τ)
CALCULATE SUBTRACTION VALUES (ϕig/Σϕig) AND (ϕib/Σϕiall) FOR ALL QoS CLASSES
1302
SUBTRACT ϕig/Σϕig FROM TOKEN VALUE (ig), SUBTRACT ϕib/Σϕiall FROM TOKEN VALUE (ib), ADD ϕig/Σϕig TO CARRIED-FORWARD (ig), AND ADD ϕib/Σϕiall TO CARRIED-FORWARD (ib)
END OF PROCESSING

FIG. 29

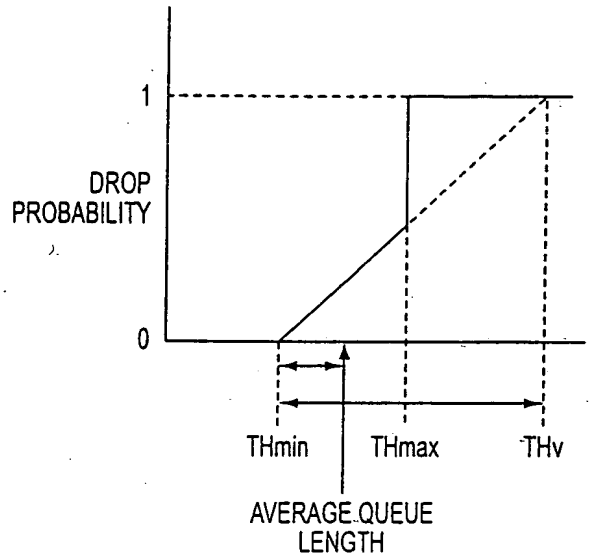


FIG. 30

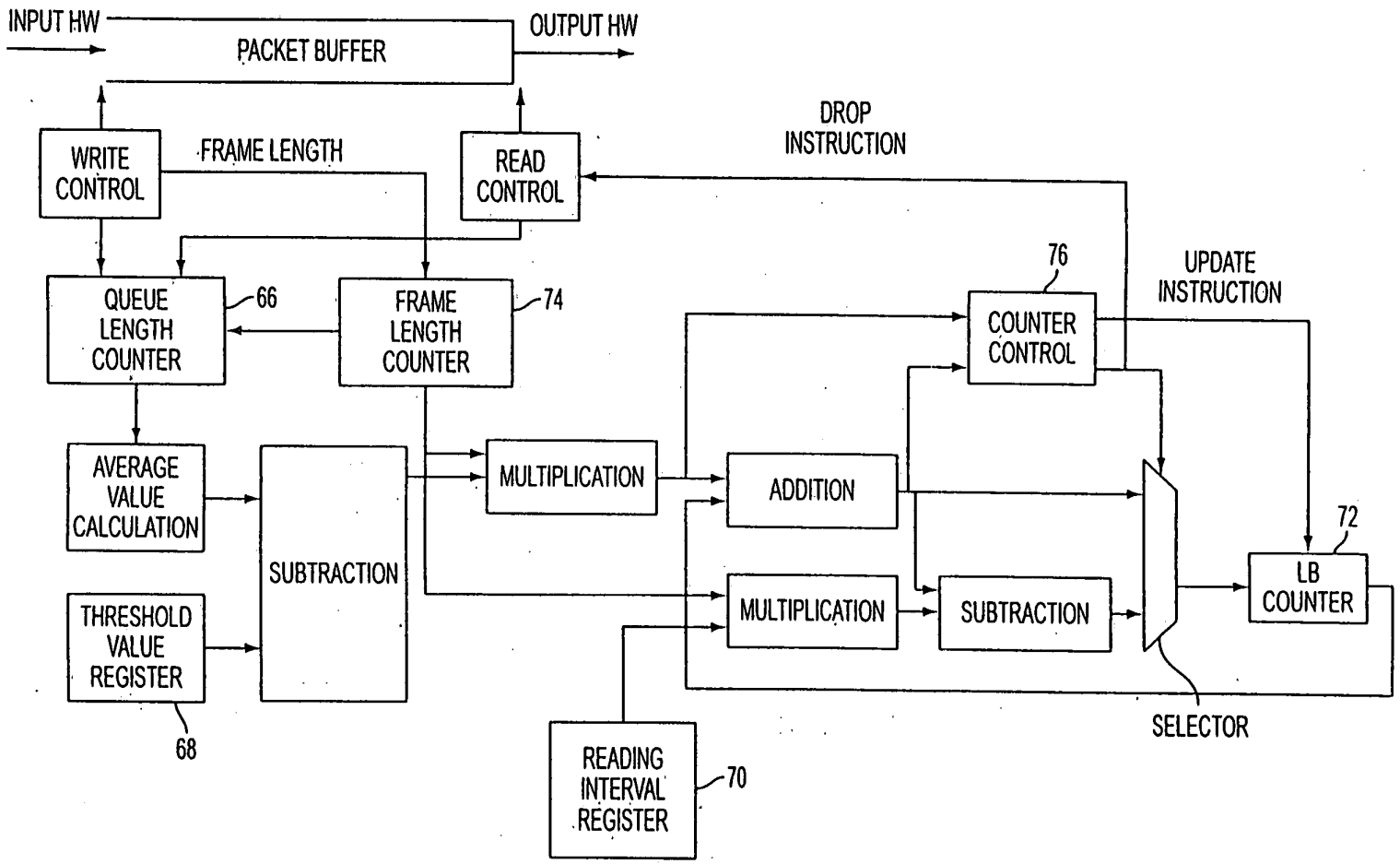
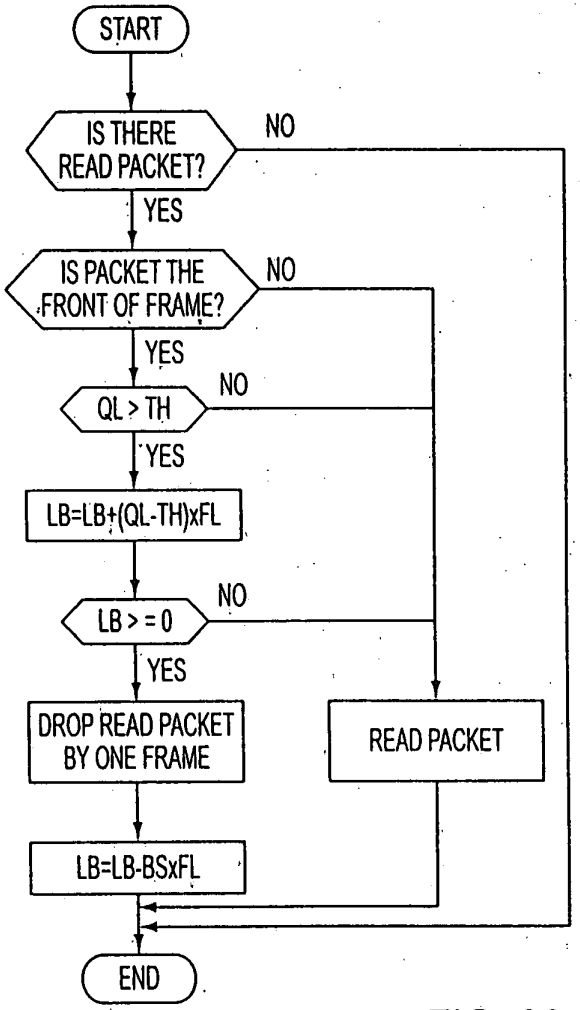

INPUT HW
PACKET BUFFER
OUTPUT HW
WRITE CONTROL
FRAME LENGTH
READ CONTROL
DROP INSTRUCTION
QUEUE LENGTH COUNTER
66
FRAME LENGTH COUNTER
74
76
COUNTER CONTROL
UPDATE INSTRUCTION
AVERAGE VALUE CALCULATION
SUBTRACTION
MULTIPLICATION
ADDITION
72
LB COUNTER
THRESHOLD VALUE REGISTER
68
MULTIPLICATION
SUBTRACTION
SELECTOR
READING INTERVAL REGISTER
70

FIG. 31



LB: LB COUNTER
QL: (AVERAGE) QUEUE LENGTH
TH: DROP THRESHOLD VALUE
FL: FRAME LENGTH
BS: READING INTERVAL

FIG. 32

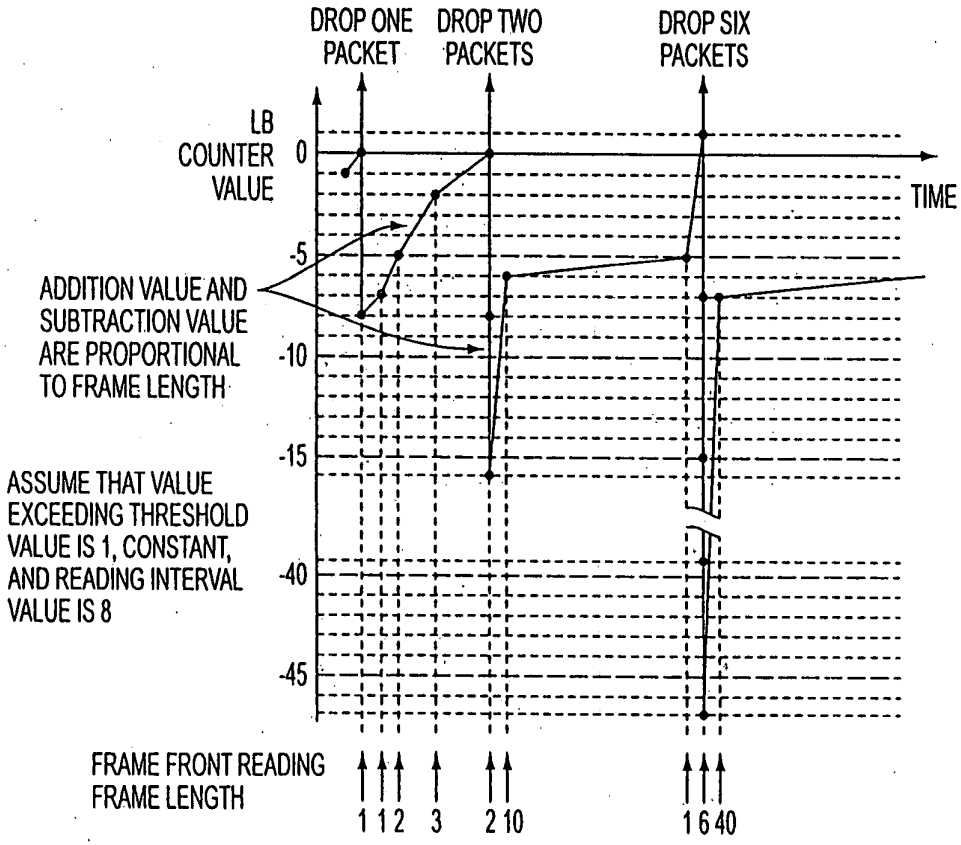
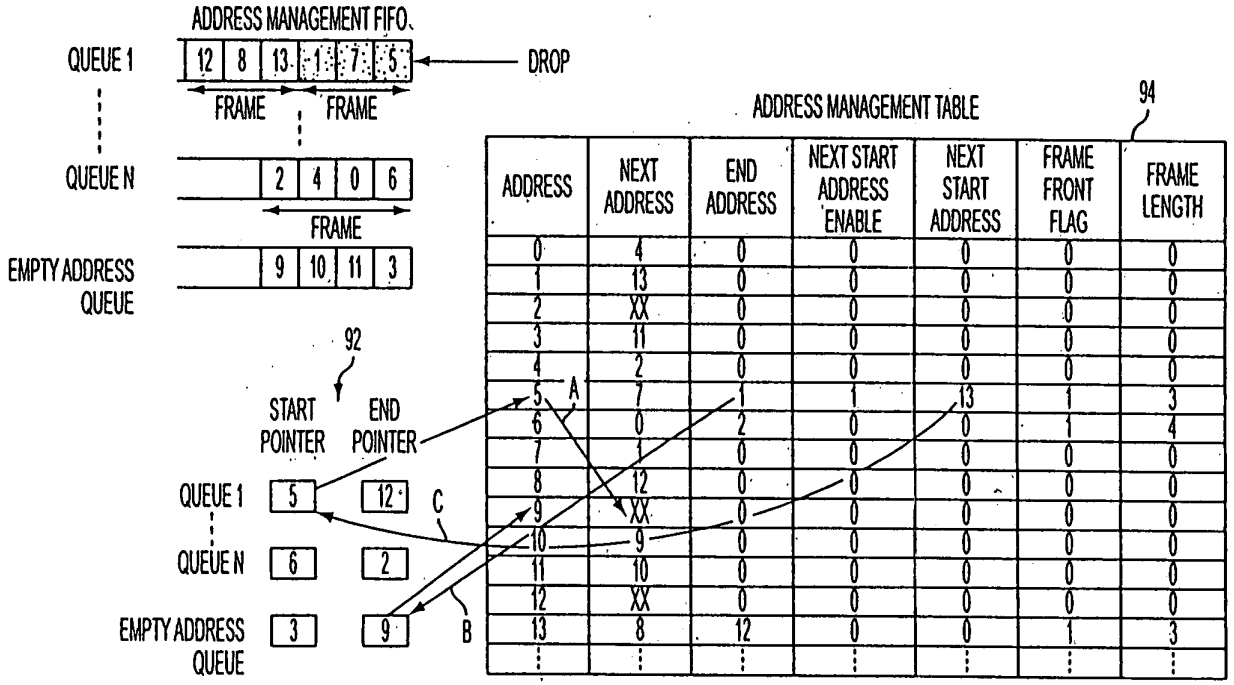

DROP ONE PACKET
DROP TWO PACKETS
DROP SIX PACKETS
LB COUNTER VALUE
0
-5
-10
-15
-40
-45
TIME
ADDITION VALUE AND SUBTRACTION VALUE ARE PROPORTIONAL TO FRAME LENGTH
ASSUME THAT VALUE EXCEEDING THRESHOLD VALUE IS 1, CONSTANT, AND READING INTERVAL VALUE IS 8
FRAME FRONT READING FRAME LENGTH
1 1 2 3 2 10 1 6 40

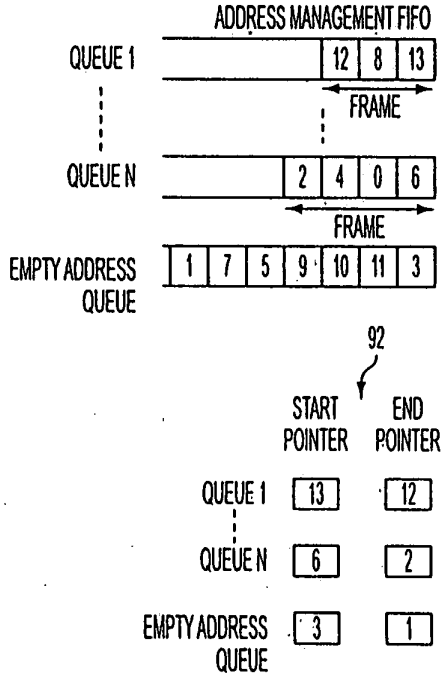
FIG. 33



ADDRESS	NEXT ADDRESS	END ADDRESS	NEXT START ADDRESS ENABLE	NEXT START ADDRESS	FRAME FRONT FLAG	FRAME LENGTH
0	4	0	0	0	0	0
1	13	0	0	0	0	0
2	XX	0	0	0	0	0
3	11	0	0	0	0	0
4	2	0	0	0	0	0
5	7	1	1	13	1	3
6	0	2	0	0	1	4
7	1	0	0	0	0	0
8	12	0	0	0	0	0
9	XX	0	0	0	0	0
10	9	0	0	0	0	0
11	10	0	0	0	0	0
12	XX	0	0	0	0	0
13	8	12	0	0	1	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮

FIG. 34

T0020 "9222860



ADDRESS MANAGEMENT TABLE 94

ADDRESS	NEXT ADDRESS	END ADDRESS	NEXT START ADDRESS ENABLE	NEXT START ADDRESS	FRAME FRONT FLAG	FRAME LENGTH
0	4	0	0	0	0	0
1	13	0	0	0	0	0
2	XX	0	0	0	0	0
3	11	0	0	0	0	0
4	2	0	0	0	0	0
5	7	0	0	0	0	0
6	0	2	0	0	1	4
7	1	0	0	0	0	0
8	12	0	0	0	0	0
9	5	0	0	0	0	0
10	9	0	0	0	0	0
11	10	0	0	0	0	0
12	XX	0	0	0	0	0
13	8	12	0	0	1	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮

FIG. 35

FOREIGN PATENT

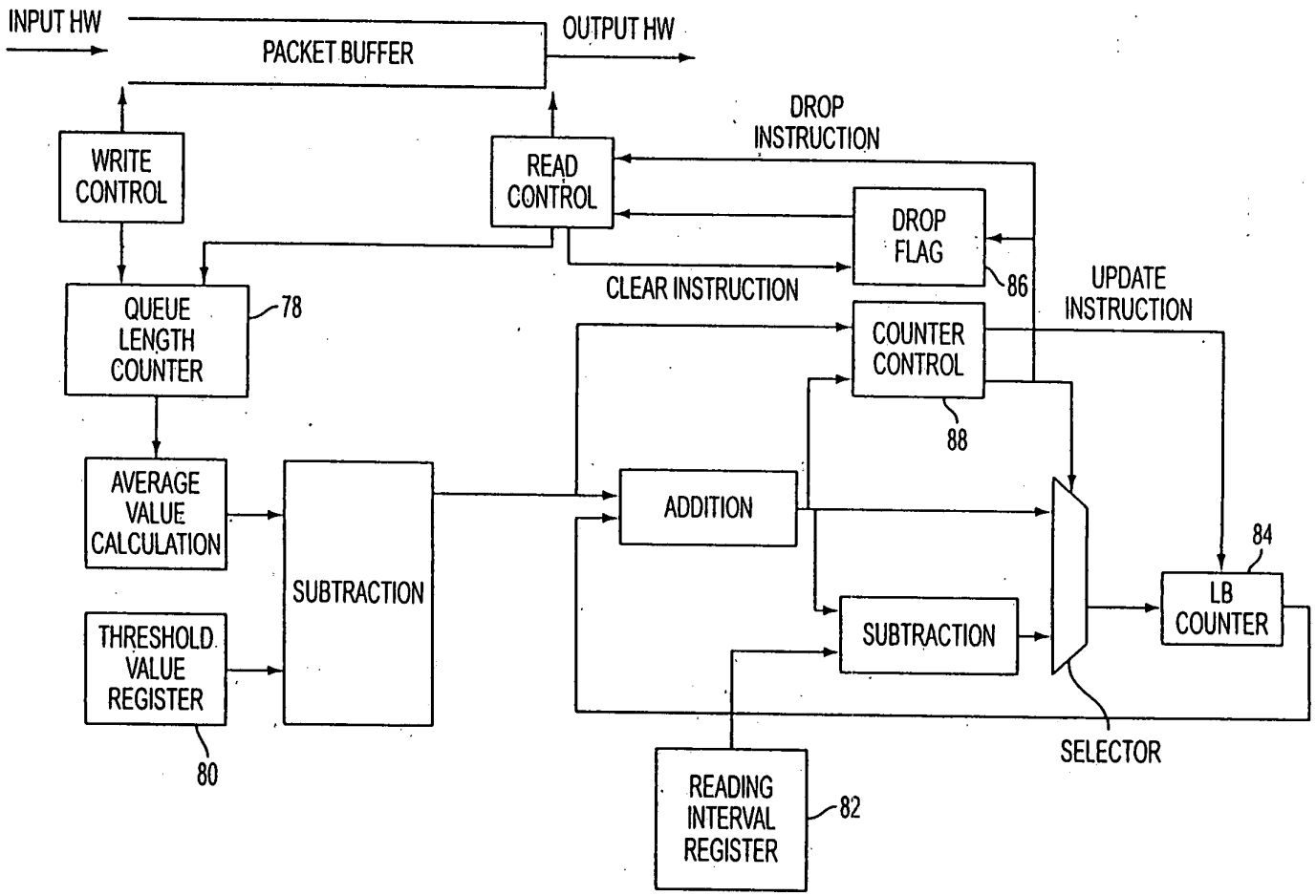


FIG. 36

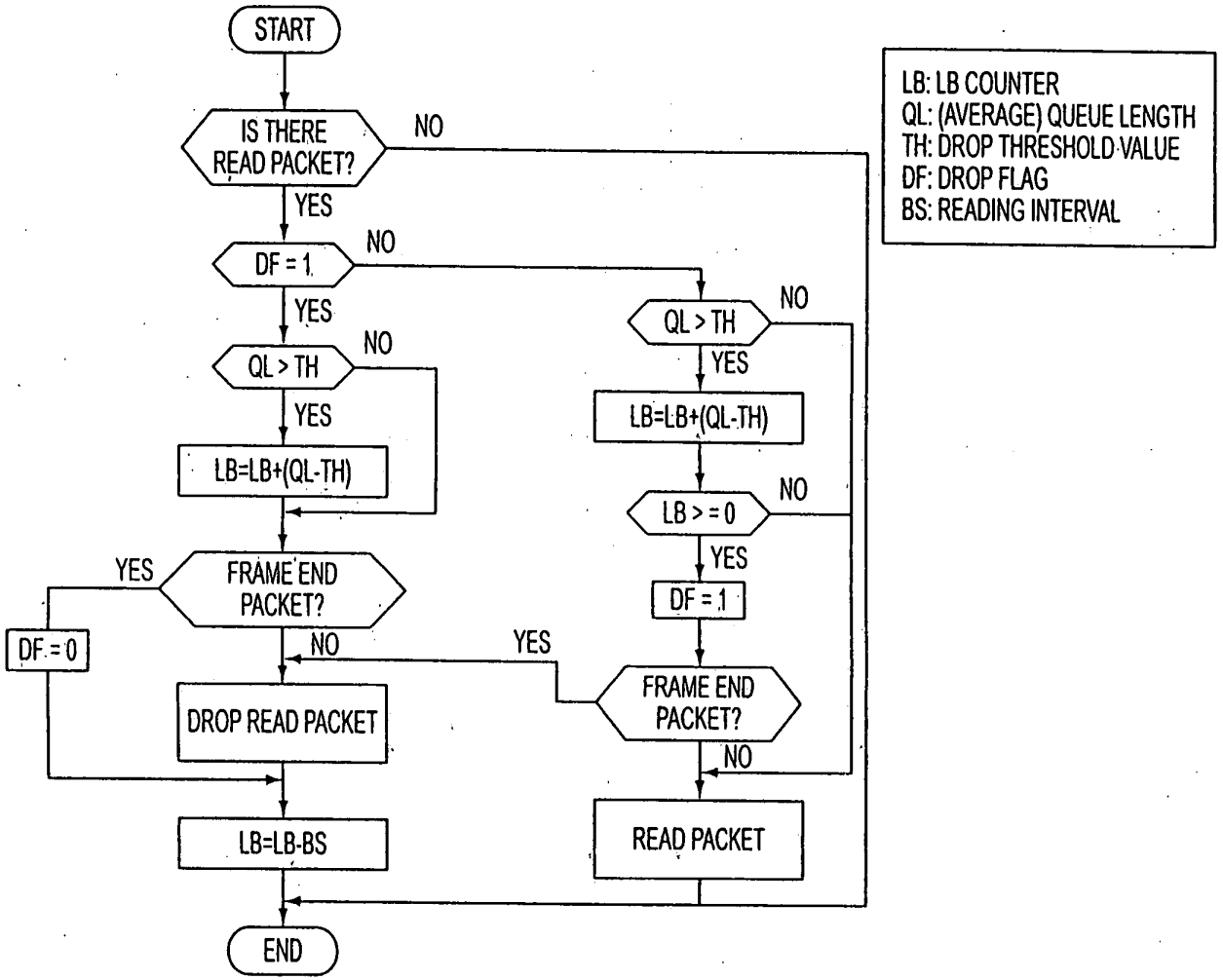

START
IS THERE READ PACKET?
NO
YES
DF = 1
NO
YES
QL > TH
NO
YES
LB=LB+(QL-TH)
YES
FRAME END PACKET?
DF = 0
NO
DROP READ PACKET
LB=LB-BS
END
QL > TH
NO
YES
LB=LB+(QL-TH)
LB > = 0
NO
YES
DF = 1
YES
FRAME END PACKET?
NO
READ PACKET
LB: LB COUNTER
QL: (AVERAGE) QUEUE LENGTH
TH: DROP THRESHOLD VALUE
DF: DROP FLAG
BS: READING INTERVAL

FIG. 37

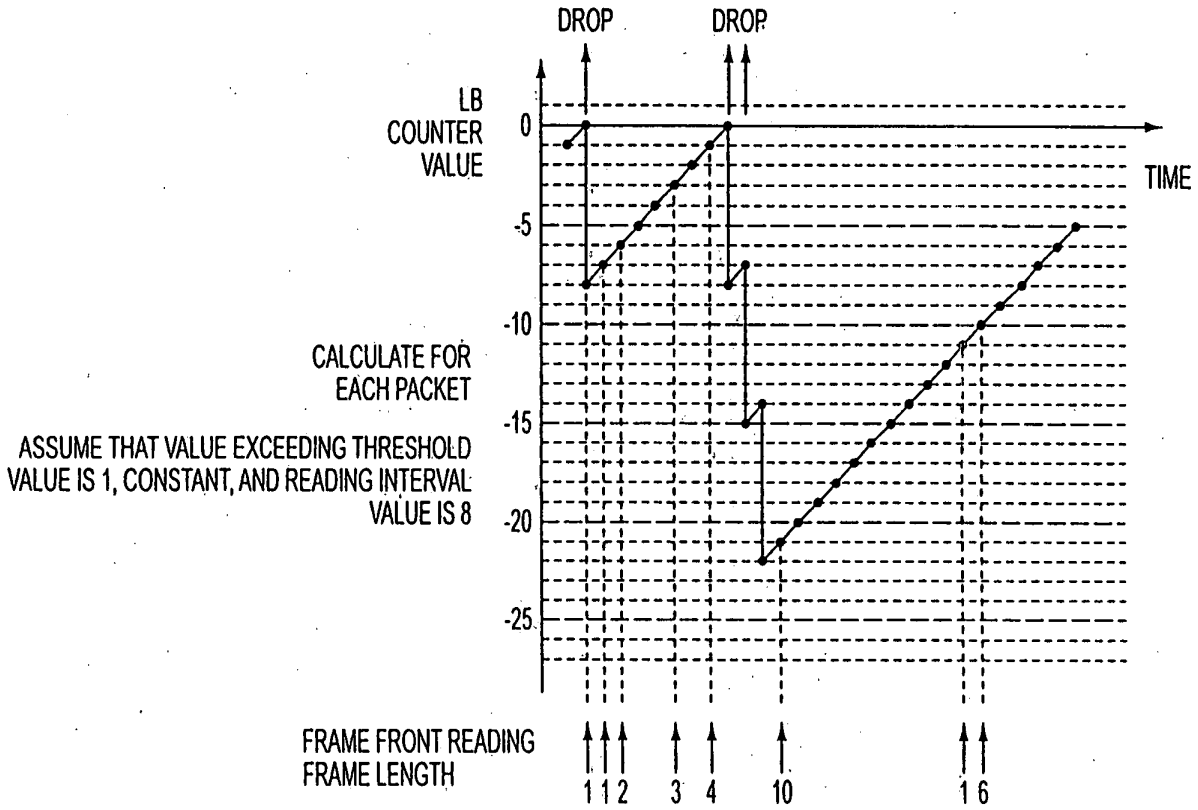

DROP
DROP
LB
COUNTER
VALUE
0
-5
-10
-15
-20
-25
TIME
CALCULATE FOR
EACH PACKET
ASSUME THAT VALUE EXCEEDING THRESHOLD
VALUE IS 1, CONSTANT, AND READING INTERVAL
VALUE IS 8
FRAME FRONT READING
FRAME LENGTH
1 1 2 3 4 10 1 6

FIG. 38

TOP20 9222860

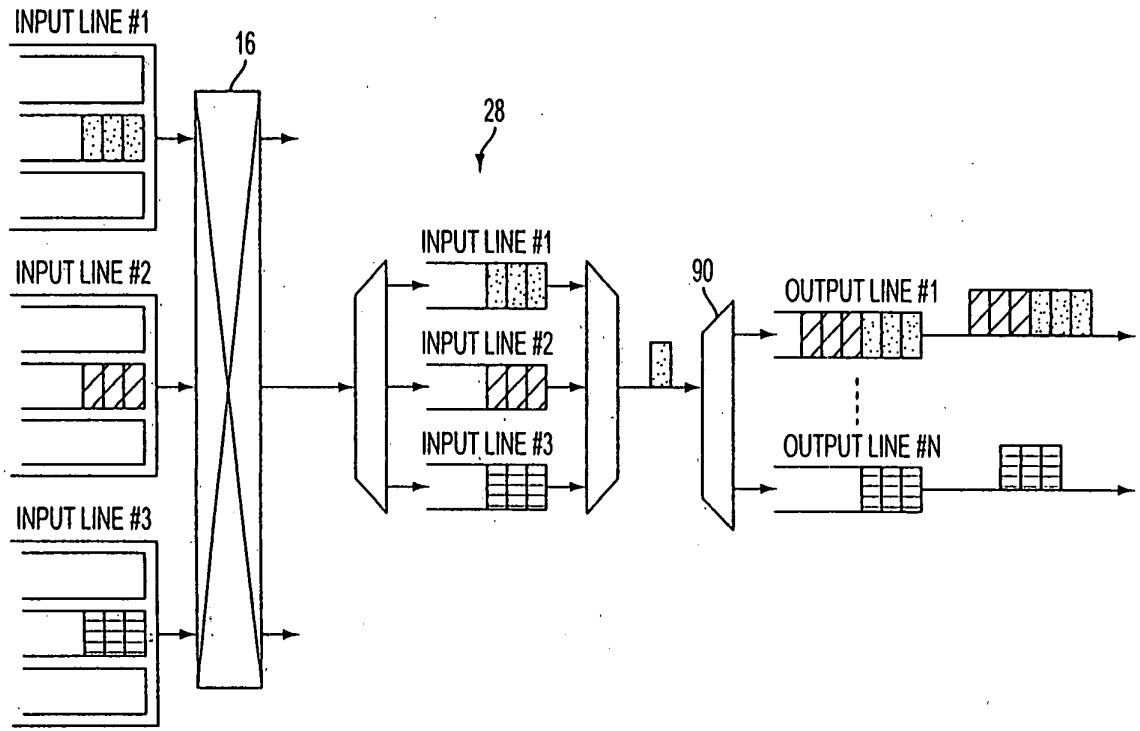


FIG. 39

FOZED 92ETGO

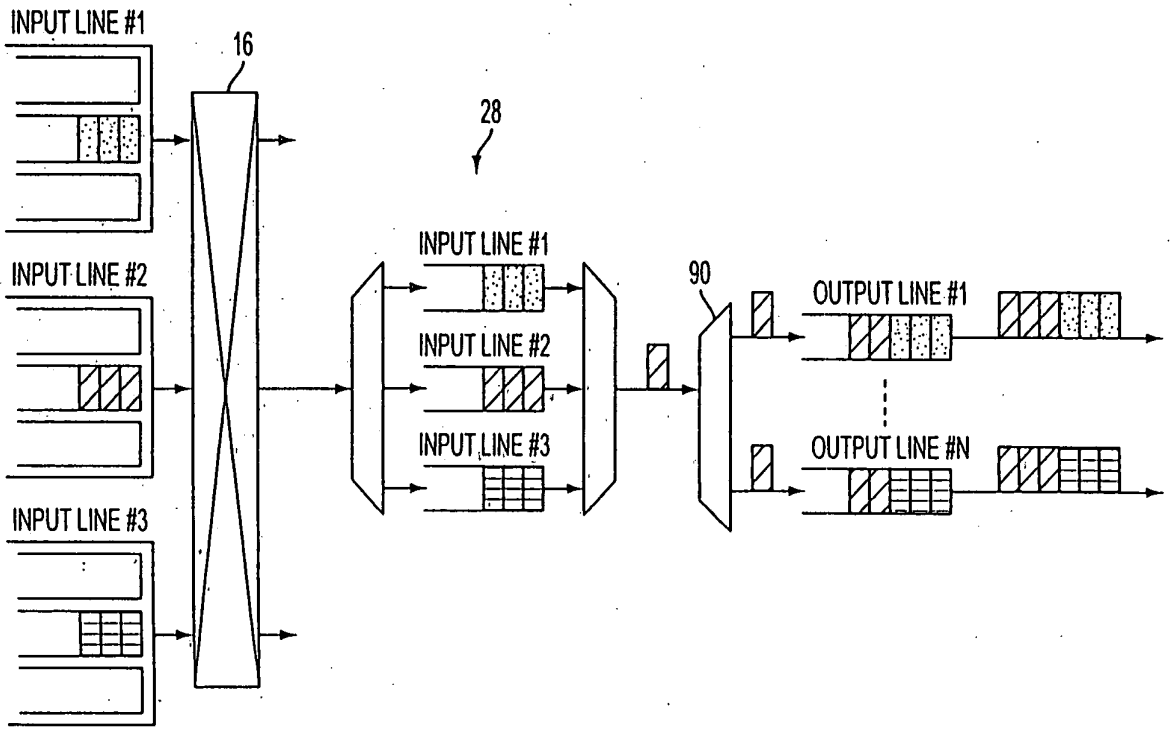


FIG. 40

T0020-922T60

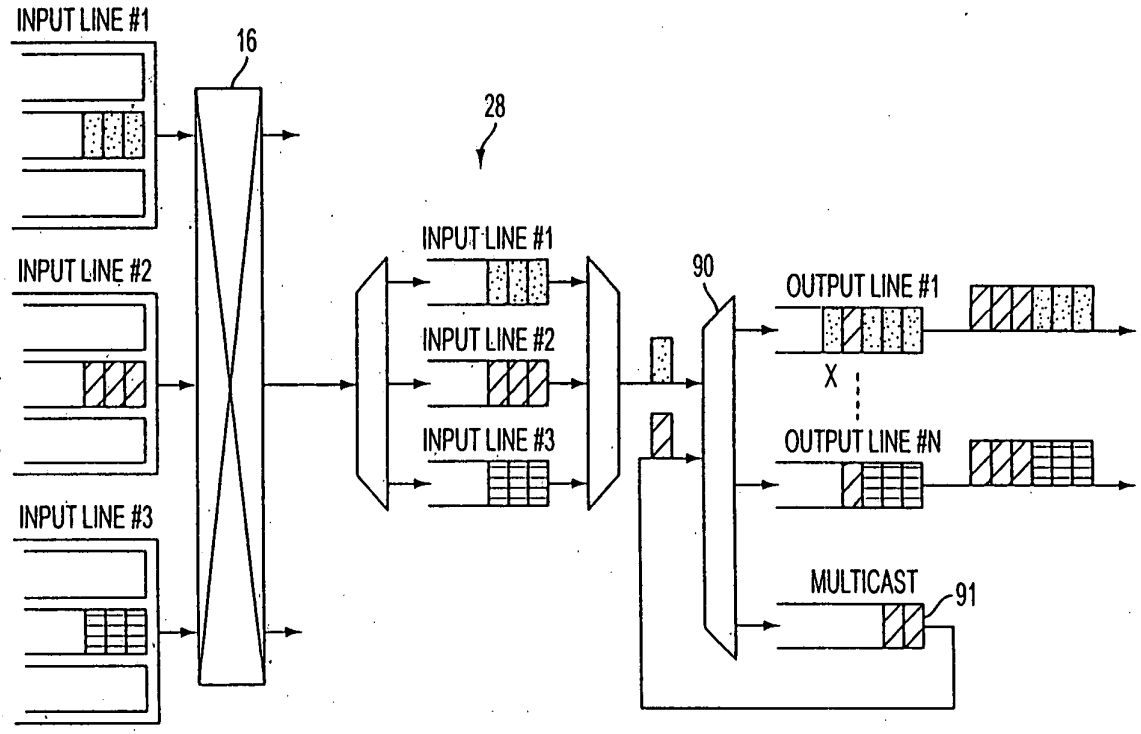


FIG. 41

T0020 9226360

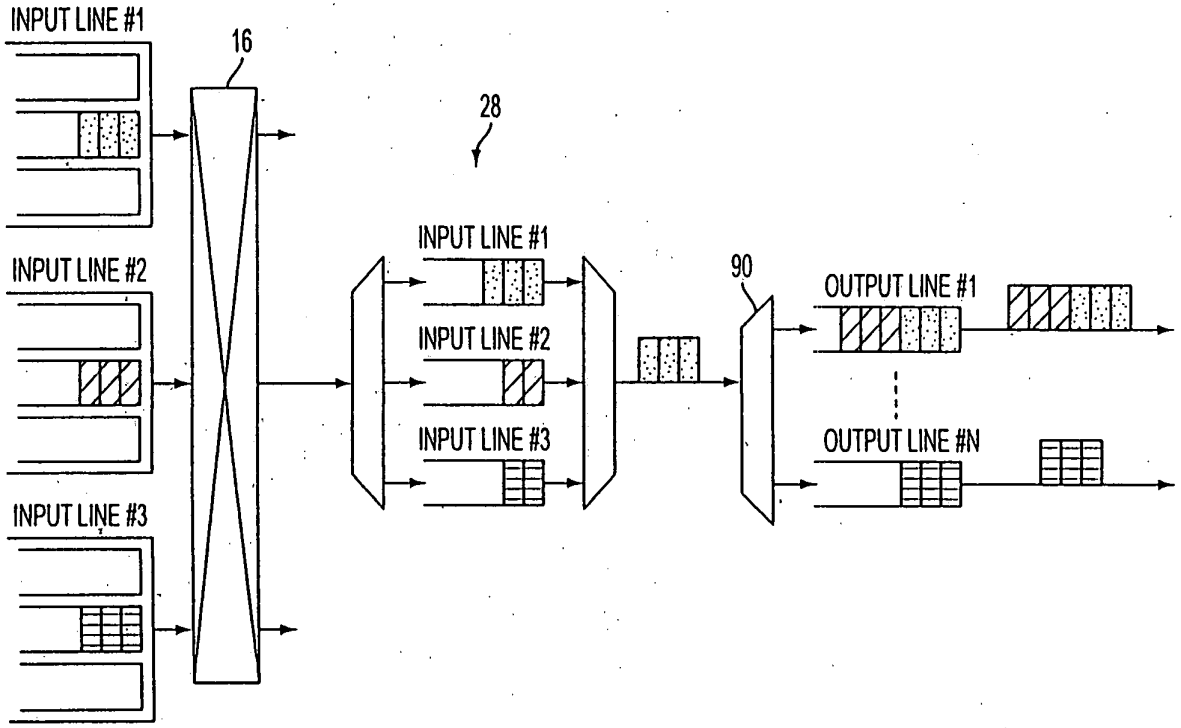


FIG. 42

F00220 922CT860

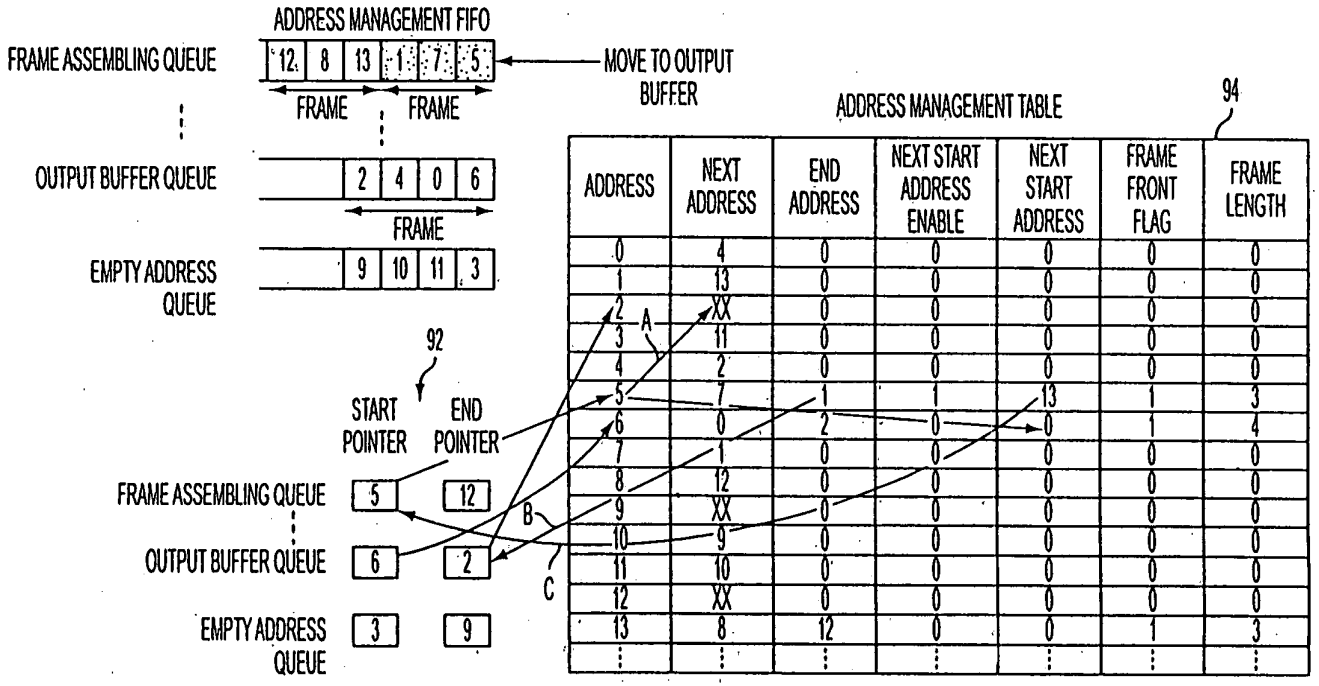
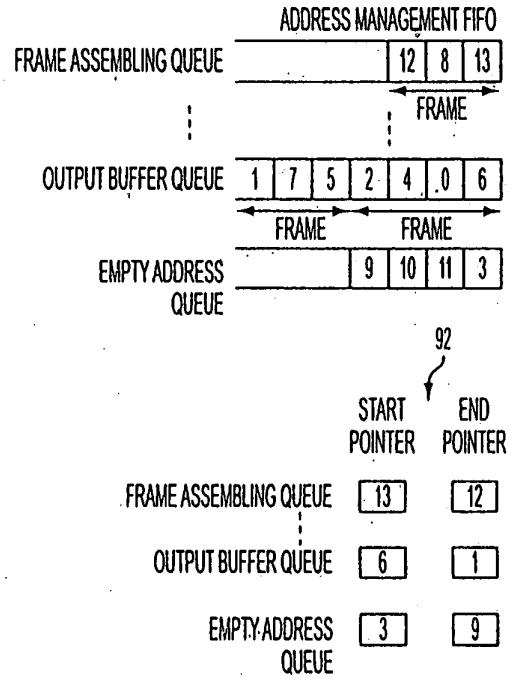


FIG. 43



ADDRESS MANAGEMENT TABLE 94

ADDRESS	NEXT ADDRESS	END ADDRESS	NEXT START ADDRESS ENABLE	NEXT START ADDRESS	FRAME FRONT FLAG	FRAME LENGTH
0	4	0	0	0	0	0
1	13	0	0	0	0	0
2	5	0	0	0	0	0
3	11	0	0	0	0	0
4	2	0	0	0	0	0
5	7	1	0	0	1	3
6	0	2	1	5	1	4
7	1	0	0	0	0	0
8	12	0	0	0	0	0
9	XX	0	0	0	0	0
10	9	0	0	0	0	0
11	10	0	0	0	0	0
12	XX	0	0	0	0	0
13	8	12	0	0	1	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮

FIG. 44

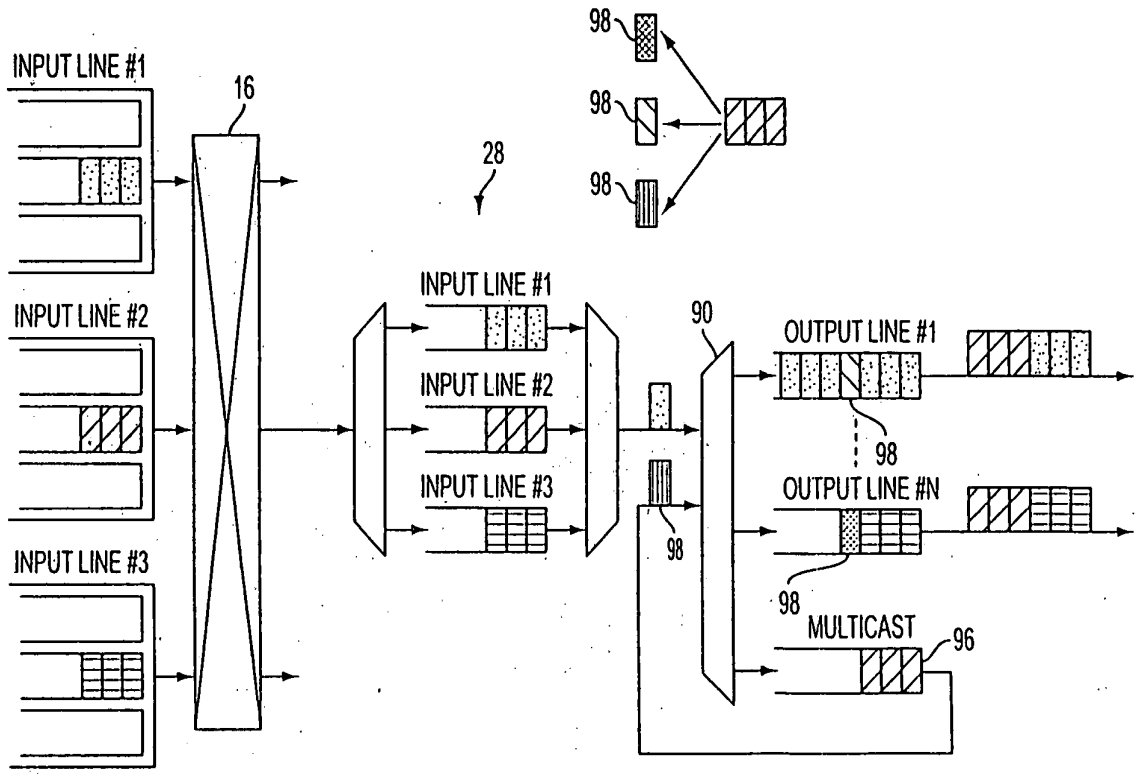

INPUT LINE #1
INPUT LINE #2
INPUT LINE #3
16
28
98
98
98
INPUT LINE #1
INPUT LINE #2
INPUT LINE #3
90
98
OUTPUT LINE #1
98
OUTPUT LINE #N
98
MULTICAST
96

FIG. 45

FOREIGN SEARCH

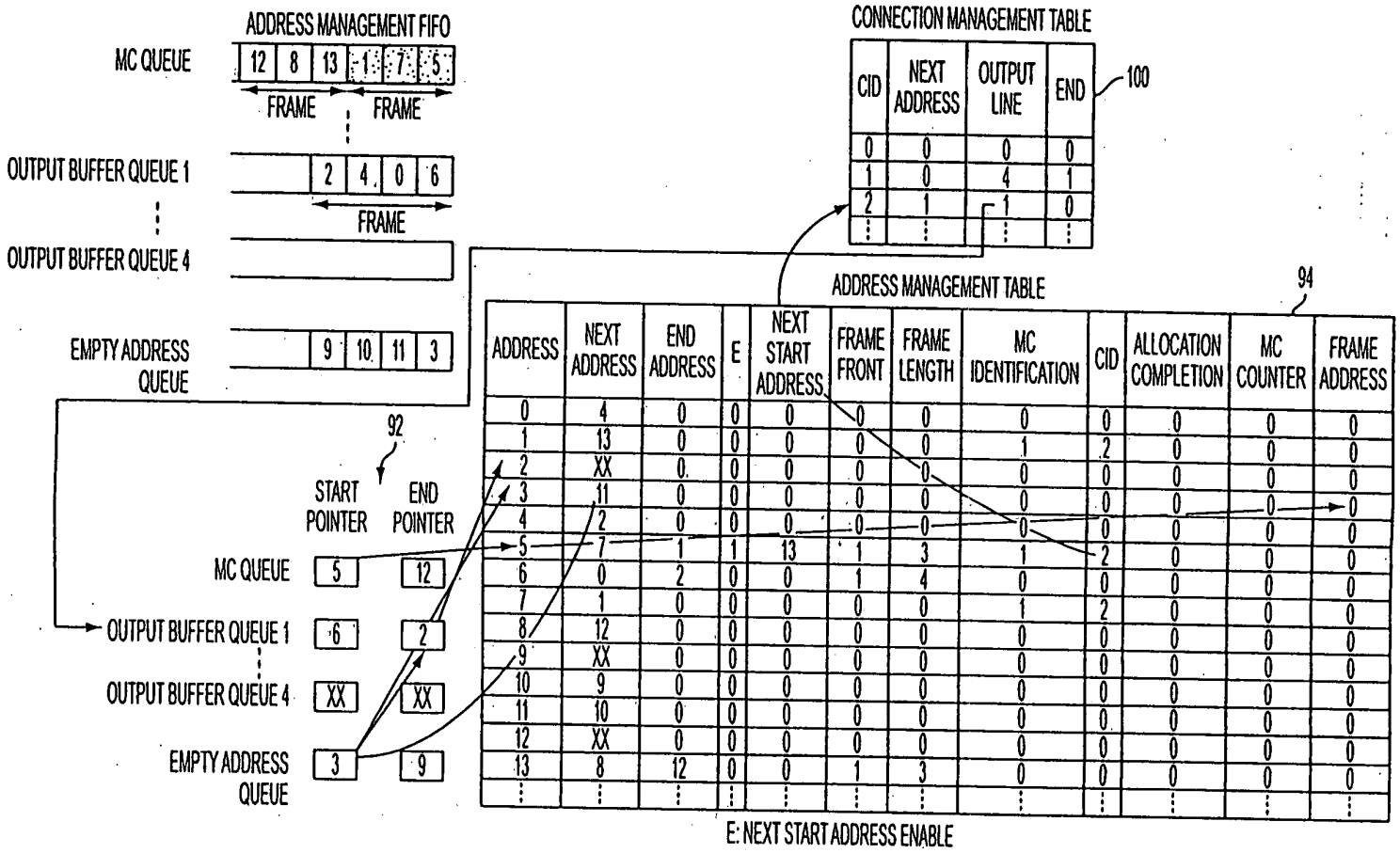
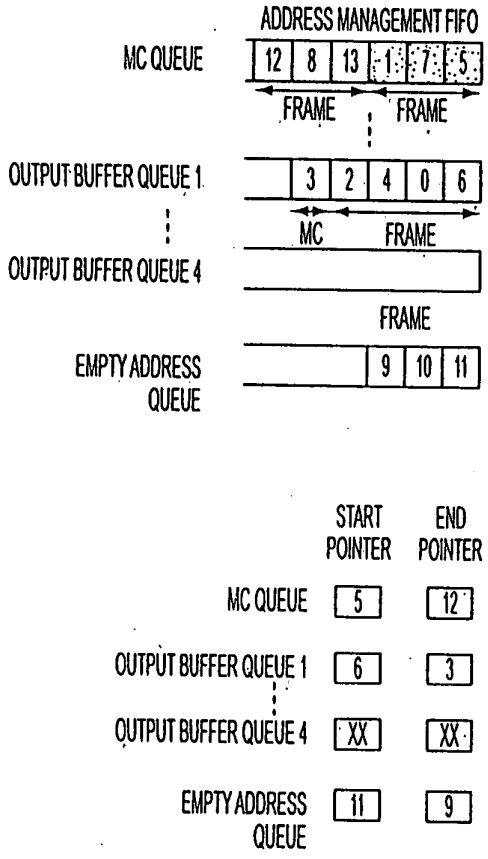


FIG. 46



CONNECTION MANAGEMENT TABLE

CID	NEXT ADDRESS	OUTPUT LINE	END
0	0	0	0
1	0	4	1
2	1	1	0
⋮	⋮	⋮	⋮

ADDRESS MANAGEMENT TABLE

ADDRESS	NEXT ADDRESS	END ADDRESS	E	NEXT START ADDRESS	FRAME FRONT	FRAME LENGTH	MC IDENTIFICATION	CID	ALLOCATION COMPLETION	MC COUNTER	FRAME ADDRESS
0	4	0	0	0	0	0	0	0	0	0	0
1	13	0	0	0	0	0	1	2	0	0	0
2	3	0	0	0	0	0	0	0	0	0	0
3	11	0	0	0	0	0	1	0	0	0	5
4	2	0	0	0	0	0	0	0	0	0	0
5	7	1	1	13	1	3	1	2	0	1	0
6	0	2	1	3	1	4	0	0	0	0	0
7	1	0	0	0	0	0	1	2	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0
9	XX	0	0	0	0	0	0	0	0	0	0
10	9	0	0	0	0	0	0	0	0	0	0
11	10	0	0	0	0	0	0	0	0	0	0
12	XX	0	0	0	0	0	0	0	0	0	0
13	8	12	0	0	1	3	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

E: NEXT START ADDRESS ENABLE

FIG. 47

FUJITSU 9E2CF660

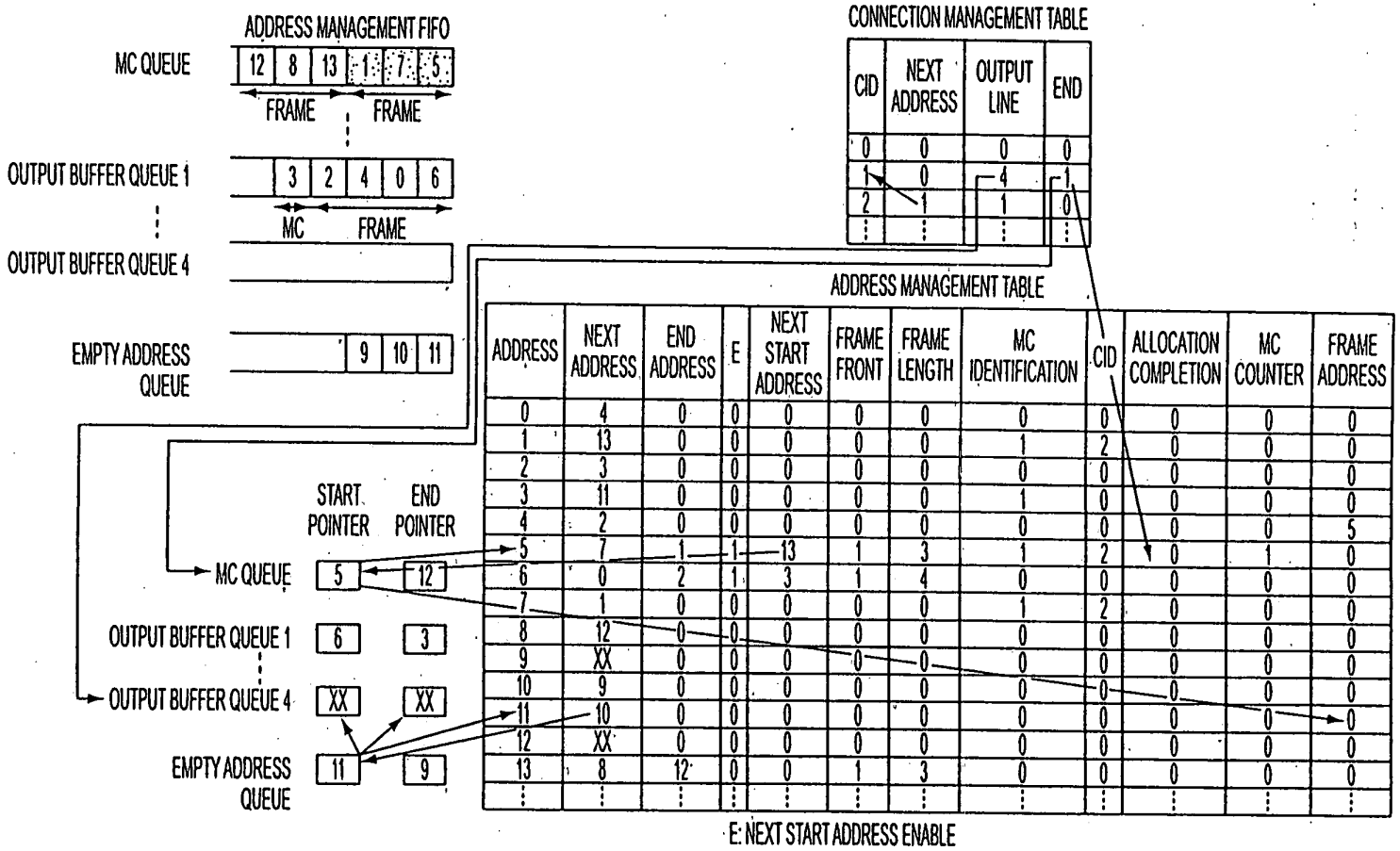
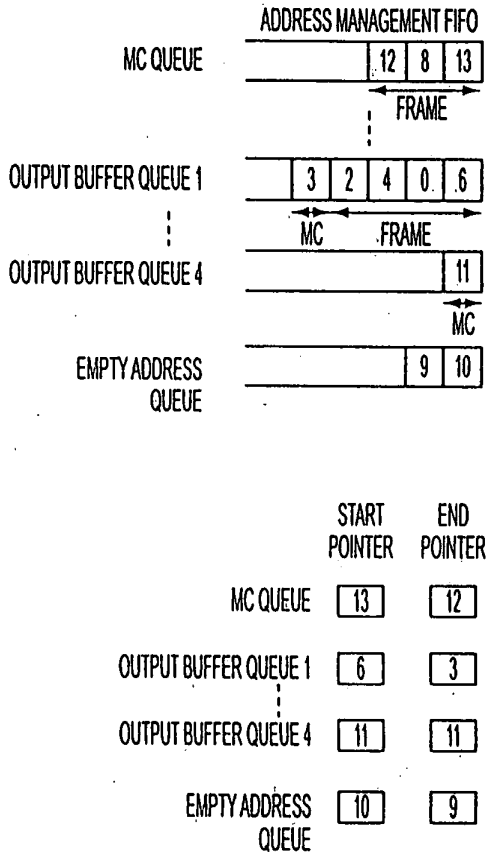


FIG. 48

T0220 "922E7860



CONNECTION MANAGEMENT TABLE

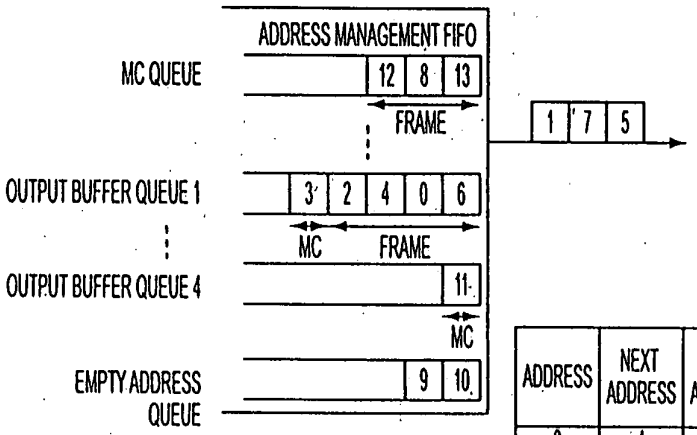
CID	NEXT ADDRESS	OUTPUT LINE	END
0	0	0	0
1	0	4	1
2	1	1	0
⋮	⋮	⋮	⋮

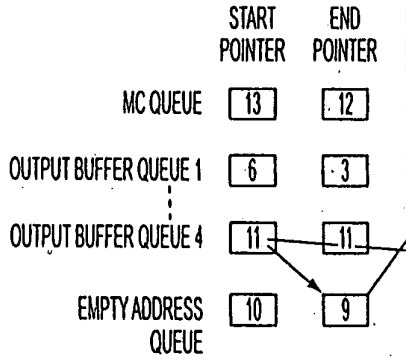
ADDRESS MANAGEMENT TABLE

ADDRESS	NEXT ADDRESS	END ADDRESS	E	NEXT START ADDRESS	FRAME FRONT	FRAME LENGTH	MC IDENTIFICATION	CID	ALLOCATION COMPLETION	MC COUNTER	FRAME ADDRESS
0	4	0	0	0	0	0	0	0	0	0	0
1	13	0	0	0	0	0	1	2	0	0	0
2	3	0	0	0	0	0	0	0	0	0	0
3	11	0	0	0	0	0	1	0	0	0	5
4	2	0	0	0	0	0	0	0	0	0	0
5	7	1	0	0	1	3	1	2	1	2	0
6	0	2	1	3	1	4	0	0	0	0	0
7	1	0	0	0	0	0	1	2	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0
9	XX	0	0	0	0	0	0	0	0	0	0
10	9	0	0	0	0	0	0	0	0	0	0
11	11	0	0	0	0	0	1	0	0	0	5
12	XX	0	0	0	0	0	0	0	0	0	0
13	8	12	0	0	1	3	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

E: NEXT START ADDRESS ENABLE

FIG. 49



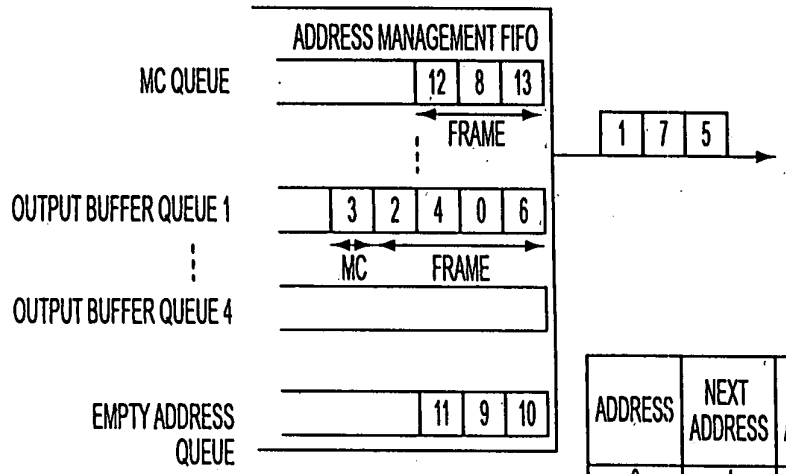


ADDRESS MANAGEMENT TABLE

ADDRESS	NEXT ADDRESS	END ADDRESS	E	NEXT START ADDRESS	FRAME FRONT	FRAME LENGTH	MC IDENTIFICATION	CID	ALLOCATION COMPLETION	MC COUNTER	FRAME ADDRESS
0	4	0	0	0	0	0	0	0	0	0	0
1	13	0	0	0	0	0	1	2	0	0	0
2	3	0	0	0	0	0	0	0	0	0	0
3	11	0	0	0	0	0	1	0	0	0	5
4	2	0	0	0	0	0	0	0	0	0	0
5	7	1	0	0	1	3	1	2	1	2	0
6	0	2	1	3	1	4	0	0	0	0	0
7	1	0	0	0	0	0	1	2	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0
9	XX	0	0	0	0	0	0	0	0	0	0
10	9	0	0	0	0	0	0	0	0	0	0
11	11	0	0	0	0	0	1	0	0	0	5
12	XX	0	0	0	0	0	0	0	0	0	0
13	8	12	0	0	1	3	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

E: NEXT START ADDRESS ENABLE

FIG. 50



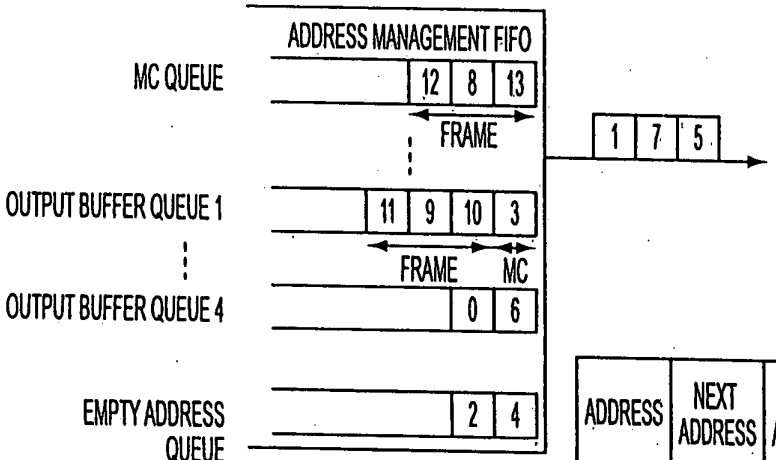
	START POINTER	END POINTER
MC QUEUE	13	12
OUTPUT BUFFER QUEUE 1	6	3
OUTPUT BUFFER QUEUE 4	11	11
EMPTY ADDRESS QUEUE	10	11

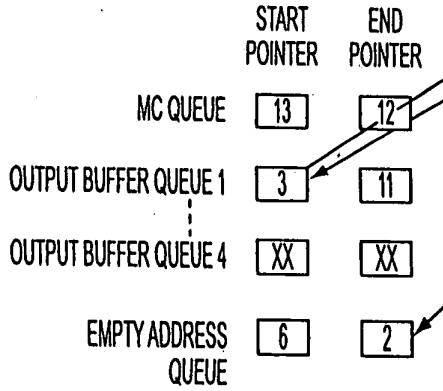
ADDRESS MANAGEMENT TABLE

ADDRESS	NEXT ADDRESS	END ADDRESS	E	NEXT START ADDRESS	FRAME FRONT	FRAME LENGTH	MC IDENTIFICATION	CID	ALLOCATION COMPLETION	MC COUNTER	FRAME ADDRESS
0	4	0	0	0	0	0	0	0	0	0	0
1	13	0	0	0	0	0	1	2	0	0	0
2	3	0	0	0	0	0	0	0	0	0	0
3	11	0	0	0	0	0	1	0	0	0	5
4	2	0	0	0	0	0	0	0	0	0	0
5	7	1	0	0	1	3	1	2	1	1	0
6	0	2	1	3	1	4	0	0	0	0	0
7	1	0	0	0	0	0	1	2	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0
9	11	0	0	0	0	0	0	0	0	0	0
10	9	0	0	0	0	0	0	0	0	0	0
11	11	0	0	0	0	0	0	0	0	0	0
12	XX	0	0	0	0	0	0	0	0	0	0
13	8	12	0	0	1	3	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

E: NEXT START ADDRESS ENABLE

FIG. 51





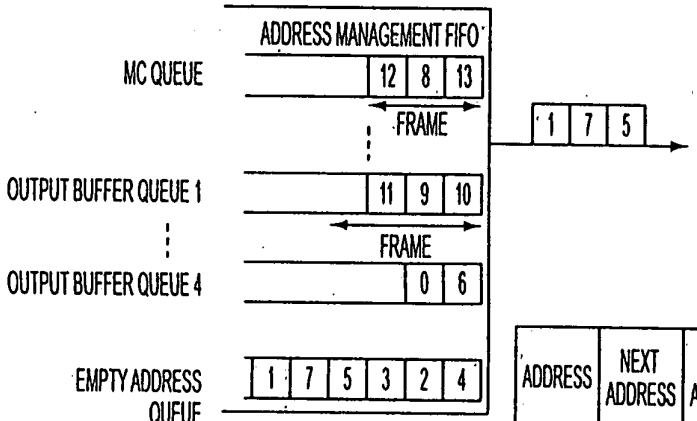
ADDRESS MANAGEMENT TABLE

ADDRESS	NEXT ADDRESS	END ADDRESS	E	NEXT START ADDRESS	FRAME FRONT	FRAME LENGTH	MC IDENTIFICATION	CID	ALLOCATION COMPLETION	MC COUNTER	FRAME ADDRESS
0	4	0	0	0	0	0	0	0	0	0	0
1	13	0	0	0	0	0	1	2	0	0	0
2	XX	0	0	0	0	0	0	0	0	0	0
3	10	0	0	0	0	0	1	0	0	0	0
4	2	0	0	0	0	0	0	0	0	0	5
5	7	1	0	0	1	3	1	2	1	1	0
6	0	2	0	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	1	2	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0
9	11	0	0	0	0	0	0	0	0	0	0
10	9	11	0	0	0	0	0	0	0	0	0
11	11	0	0	0	0	0	0	0	0	0	0
12	XX	0	0	0	0	0	0	0	0	0	0
13	8	12	0	0	1	3	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

E: NEXT START ADDRESS ENABLE

FIG. 52

T0020 "922C1860



	START POINTER	END POINTER
MC QUEUE	13	12
OUTPUT BUFFER QUEUE 1	3	11
OUTPUT BUFFER QUEUE 4	XX	XX
EMPTY ADDRESS QUEUE	6	1

ADDRESS MANAGEMENT TABLE

ADDRESS	NEXT ADDRESS	END ADDRESS	E	NEXT START ADDRESS	FRAME FRONT	FRAME LENGTH	MC IDENTIFICATION	CID	ALLOCATION COMPLETION	MC COUNTER	FRAME ADDRESS
0	4	0	0	0	0	0	0	0	0	0	0
1	13	0	0	0	0	0	1	2	0	0	0
2	3	0	0	0	0	0	0	0	0	0	0
3	5	0	0	0	0	0	1	0	0	0	5
4	2	0	0	0	0	0	0	0	0	0	0
5	7	0	0	0	0	0	0	2	0	0	0
6	0	2	0	0	0	0	0	0	0	0	0
7	1	0	0	0	0	0	1	2	0	0	0
8	12	0	0	0	0	0	0	0	0	0	0
9	11	0	0	0	0	0	0	0	0	0	0
10	9	11	0	0	0	0	0	0	0	0	0
11	11	0	0	0	0	0	0	0	0	0	0
12	XX	0	0	0	0	0	0	0	0	0	0
13	8	12	0	0	1	3	0	0	0	0	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

E: NEXT START ADDRESS ENABLE

FIG. 53

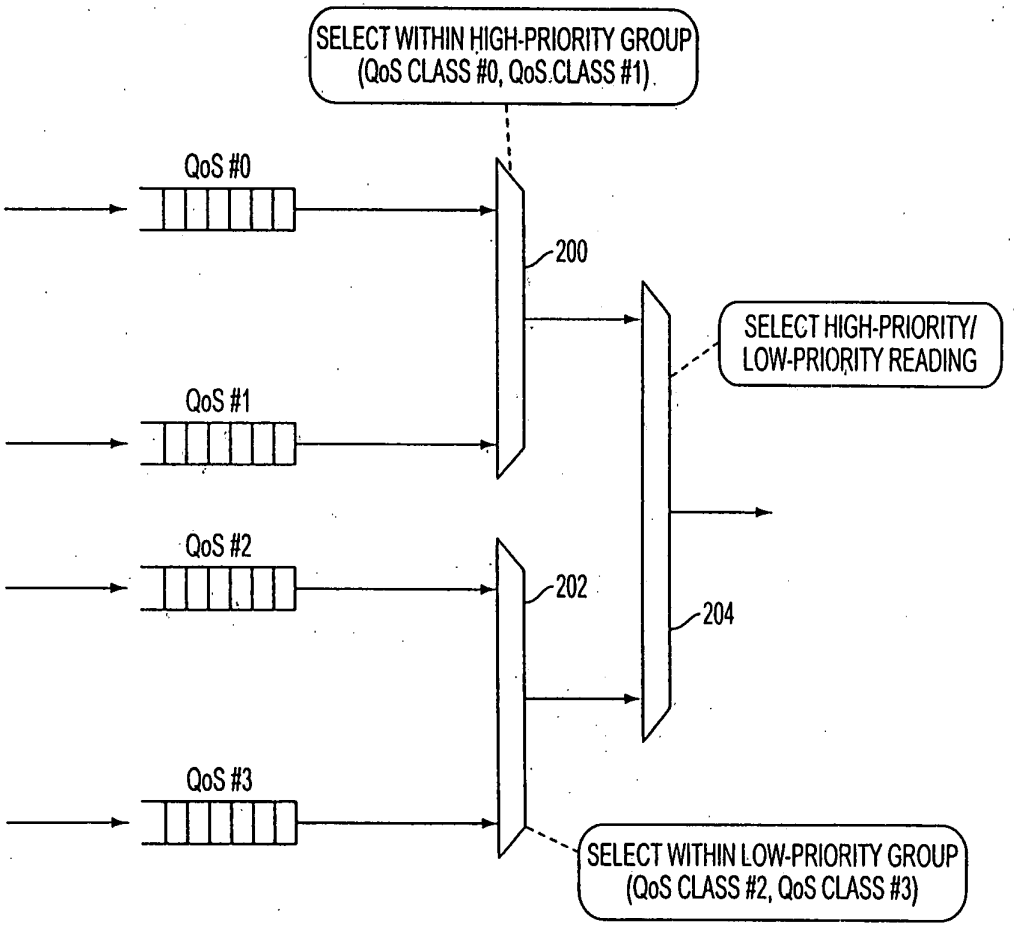

SELECT WITHIN HIGH-PRIORITY GROUP
(QoS CLASS #0, QoS CLASS #1)
QoS #0
QoS #1
QoS #2
QoS #3
200
202
204
SELECT HIGH-PRIORITY/
LOW-PRIORITY READING
SELECT WITHIN LOW-PRIORITY GROUP
(QoS CLASS #2, QoS CLASS #3)

FIG. 54

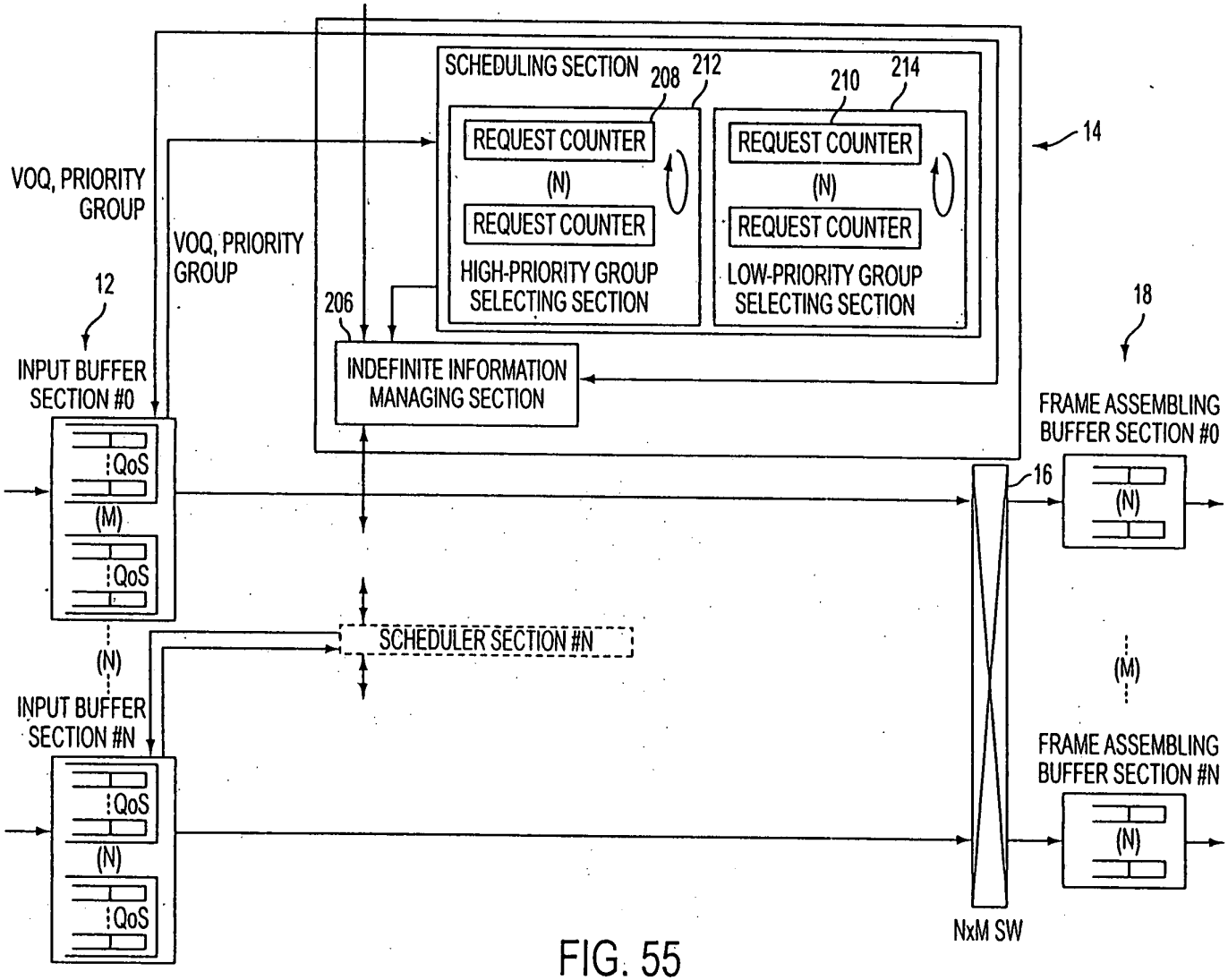

SCHEDULING SECTION
208
212
210
214
REQUEST COUNTER
(N)
REQUEST COUNTER
HIGH-PRIORITY GROUP SELECTING SECTION
REQUEST COUNTER
(N)
REQUEST COUNTER
LOW-PRIORITY GROUP SELECTING SECTION
14
VOQ, PRIORITY GROUP
VOQ, PRIORITY GROUP
12
206
INDEFINITE INFORMATION MANAGING SECTION
18
INPUT BUFFER SECTION #0
FRAME ASSEMBLING BUFFER SECTION #0
16
QoS
(M)
QoS
(N)
SCHEDULER SECTION #N
(M)
INPUT BUFFER SECTION #N
FRAME ASSEMBLING BUFFER SECTION #N
QoS
(N)
QoS
(N)
NxM SW

FIG. 55

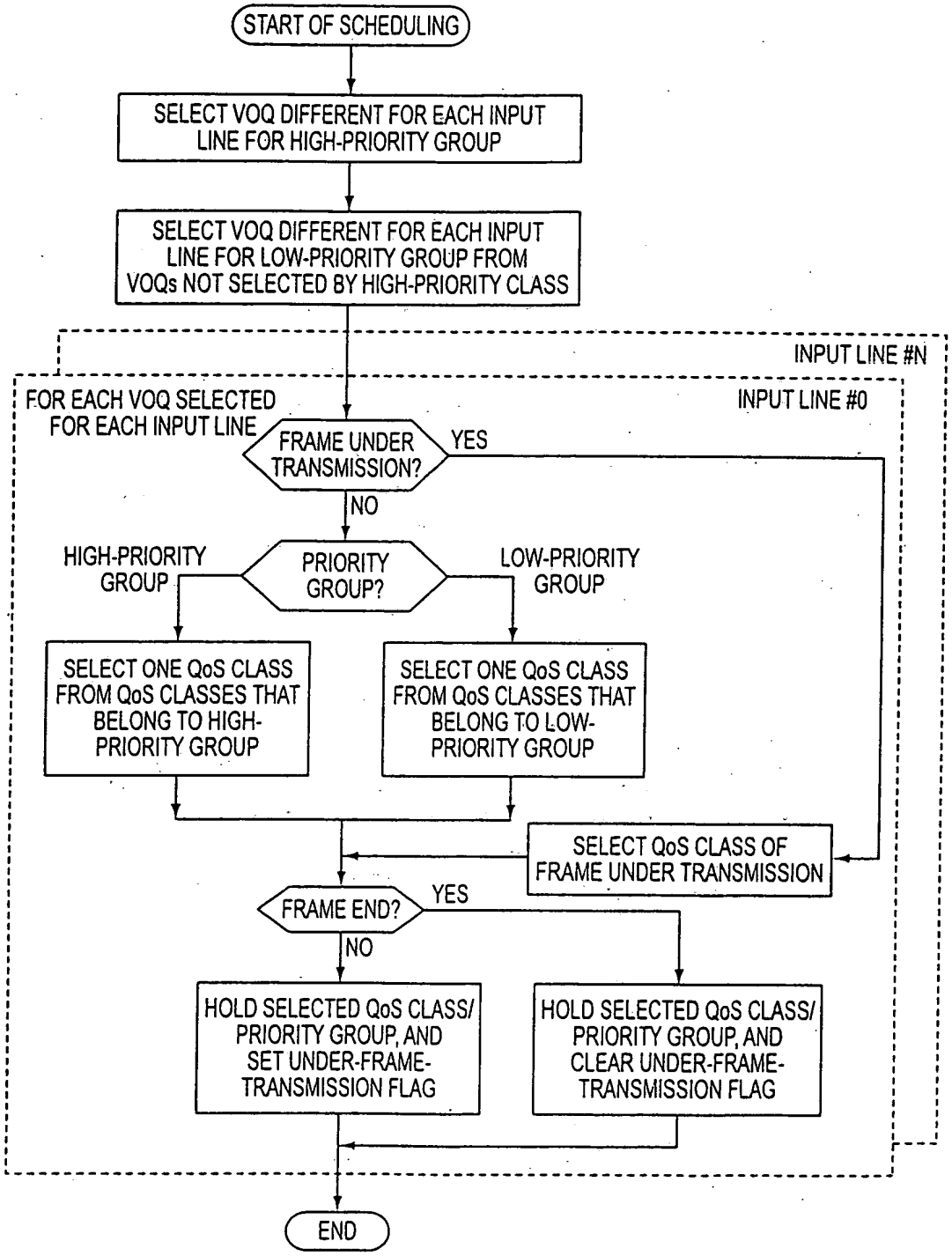

START OF SCHEDULING
SELECT VOQ DIFFERENT FOR EACH INPUT LINE FOR HIGH-PRIORITY GROUP
SELECT VOQ DIFFERENT FOR EACH INPUT LINE FOR LOW-PRIORITY GROUP FROM VOQs NOT SELECTED BY HIGH-PRIORITY CLASS
INPUT LINE #N
INPUT LINE #0
FOR EACH VOQ SELECTED FOR EACH INPUT LINE
FRAME UNDER TRANSMISSION?
YES
NO
HIGH-PRIORITY GROUP
PRIORITY GROUP?
LOW-PRIORITY GROUP
SELECT ONE QoS CLASS FROM QoS CLASSES THAT BELONG TO HIGH-PRIORITY GROUP
SELECT ONE QoS CLASS FROM QoS CLASSES THAT BELONG TO LOW-PRIORITY GROUP
SELECT QoS CLASS OF FRAME UNDER TRANSMISSION
FRAME END?
YES
NO
HOLD SELECTED QoS CLASS/ PRIORITY GROUP, AND SET UNDER-FRAME-TRANSMISSION FLAG
HOLD SELECTED QoS CLASS/ PRIORITY GROUP, AND CLEAR UNDER-FRAME-TRANSMISSION FLAG
END

FIG. 56

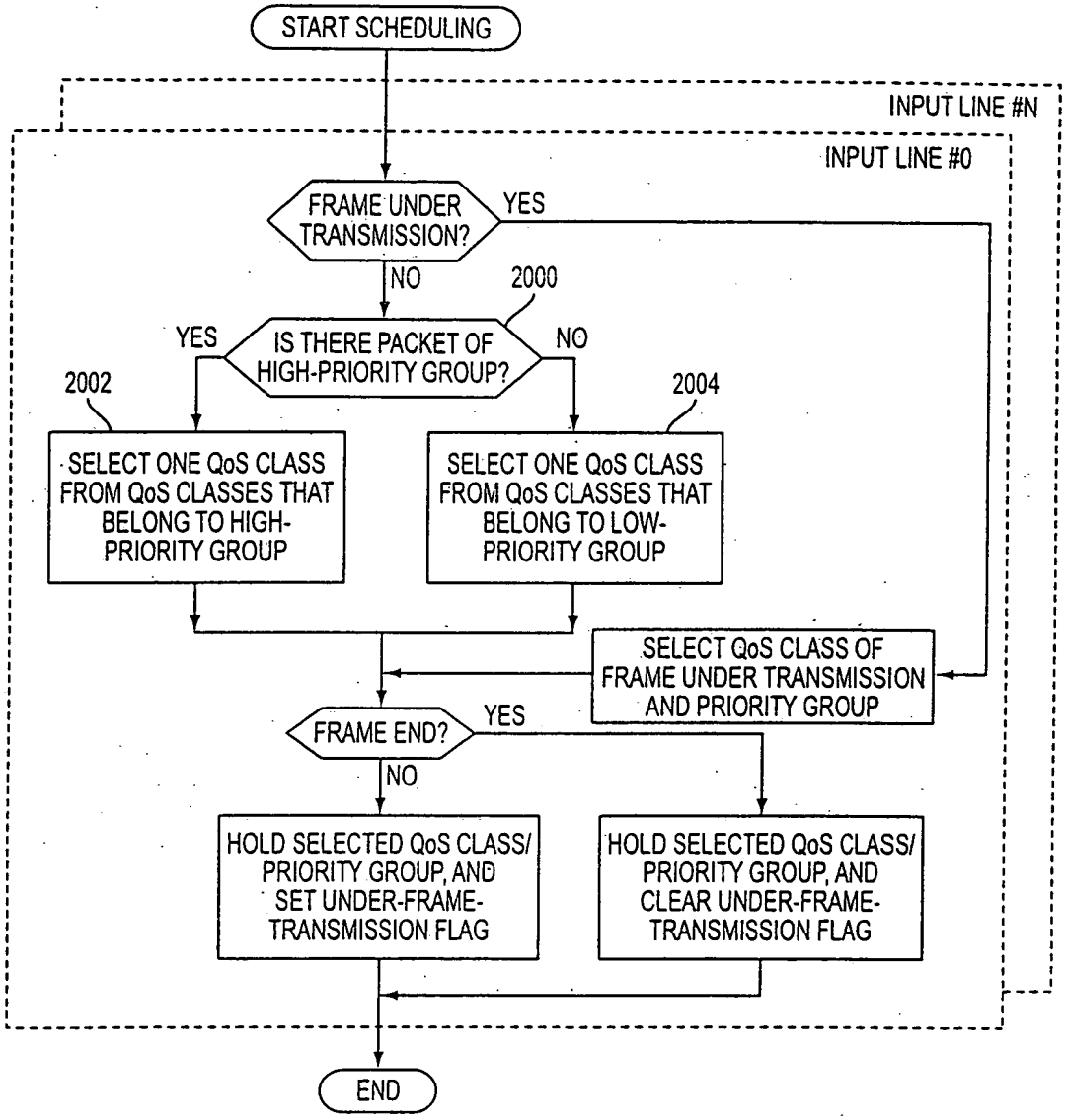

START SCHEDULING
INPUT LINE #N
INPUT LINE #0
FRAME UNDER TRANSMISSION?
YES
NO
2000
IS THERE PACKET OF HIGH-PRIORITY GROUP?
YES
NO
2002
2004
SELECT ONE QoS CLASS FROM QoS CLASSES THAT BELONG TO HIGH-PRIORITY GROUP
SELECT ONE QoS CLASS FROM QoS CLASSES THAT BELONG TO LOW-PRIORITY GROUP
SELECT QoS CLASS OF FRAME UNDER TRANSMISSION AND PRIORITY GROUP
FRAME END?
YES
NO
HOLD SELECTED QoS CLASS/ PRIORITY GROUP, AND SET UNDER-FRAME-TRANSMISSION FLAG
HOLD SELECTED QoS CLASS/ PRIORITY GROUP, AND CLEAR UNDER-FRAME-TRANSMISSION FLAG
END

FIG. 57

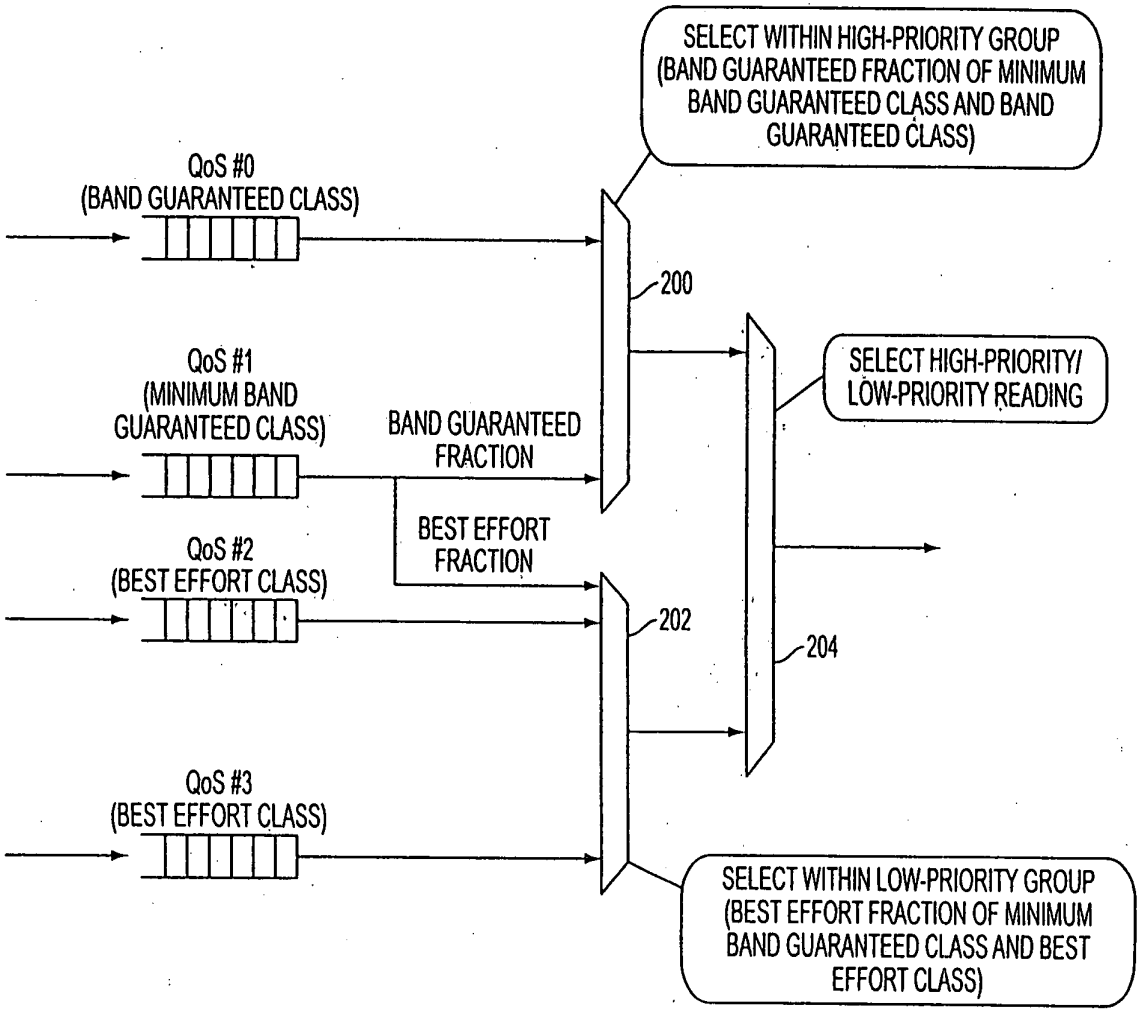

SELECT WITHIN HIGH-PRIORITY GROUP (BAND GUARANTEED FRACTION OF MINIMUM BAND GUARANTEED CLASS AND BAND GUARANTEED CLASS)
QoS #0 (BAND GUARANTEED CLASS)
200
QoS #1 (MINIMUM BAND GUARANTEED CLASS)
BAND GUARANTEED FRACTION
SELECT HIGH-PRIORITY/ LOW-PRIORITY READING
BEST EFFORT FRACTION
QoS #2 (BEST EFFORT CLASS)
202
204
QoS #3 (BEST EFFORT CLASS)
SELECT WITHIN LOW-PRIORITY GROUP (BEST EFFORT FRACTION OF MINIMUM BAND GUARANTEED CLASS AND BEST EFFORT CLASS)

FIG. 58

TOD26D' 922ET86D

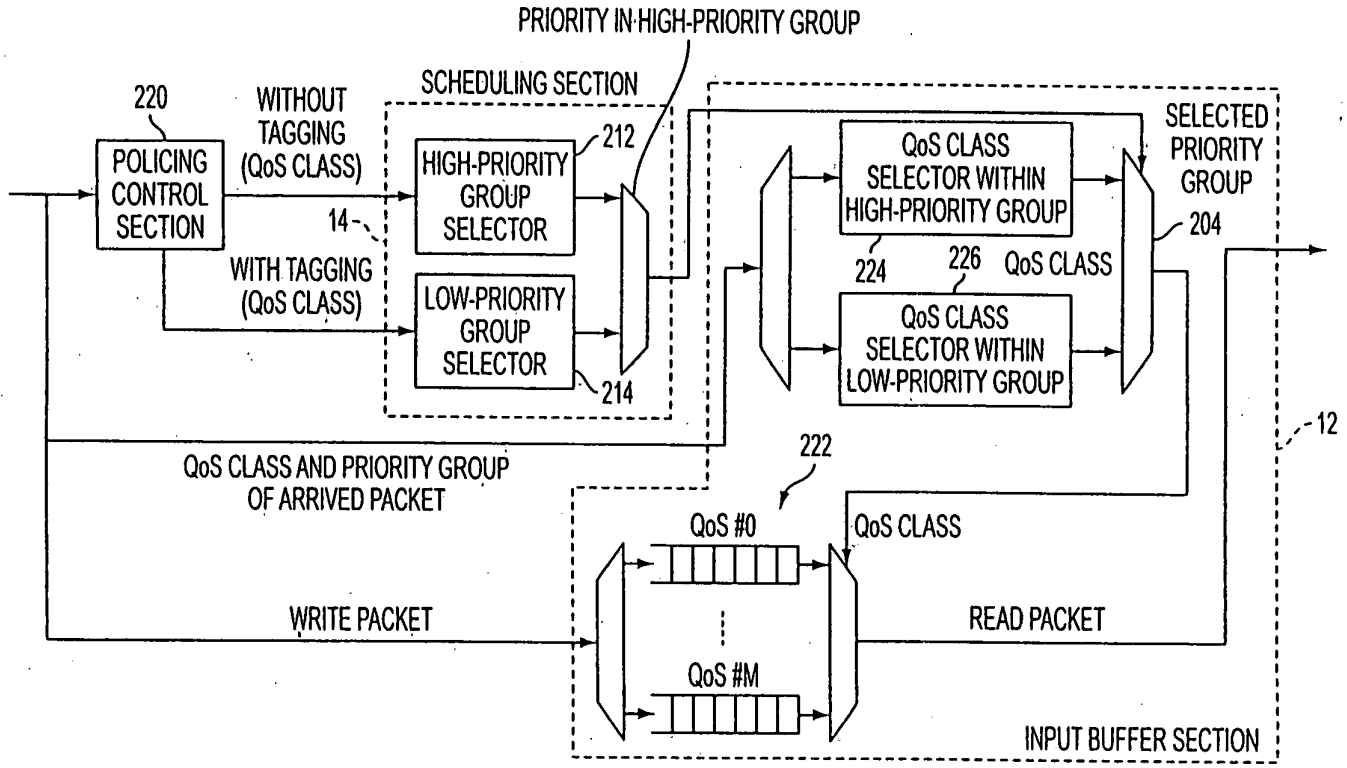


FIG. 59

T00220 "9227260

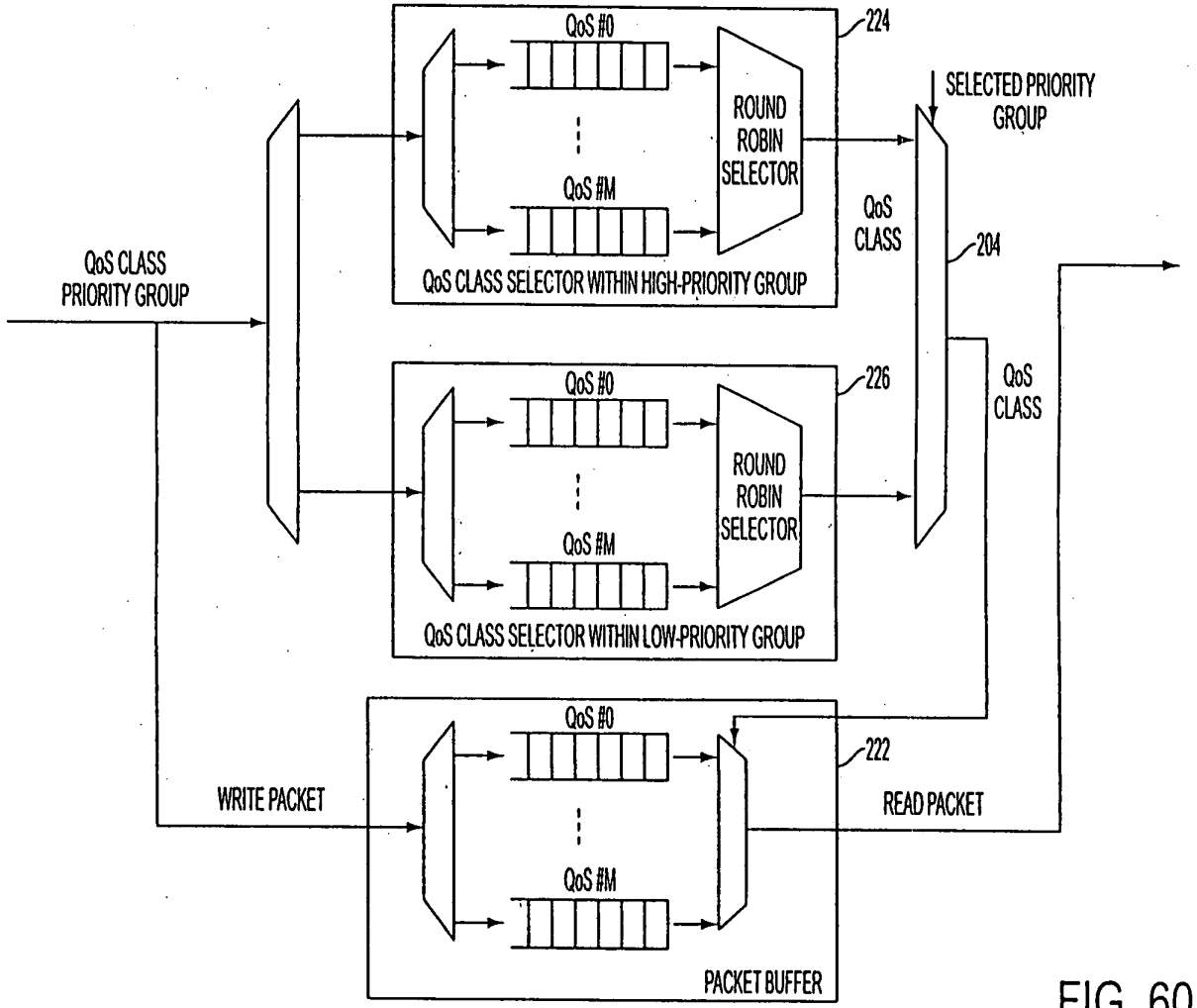


FIG. 60

TOSZO 9222T86O

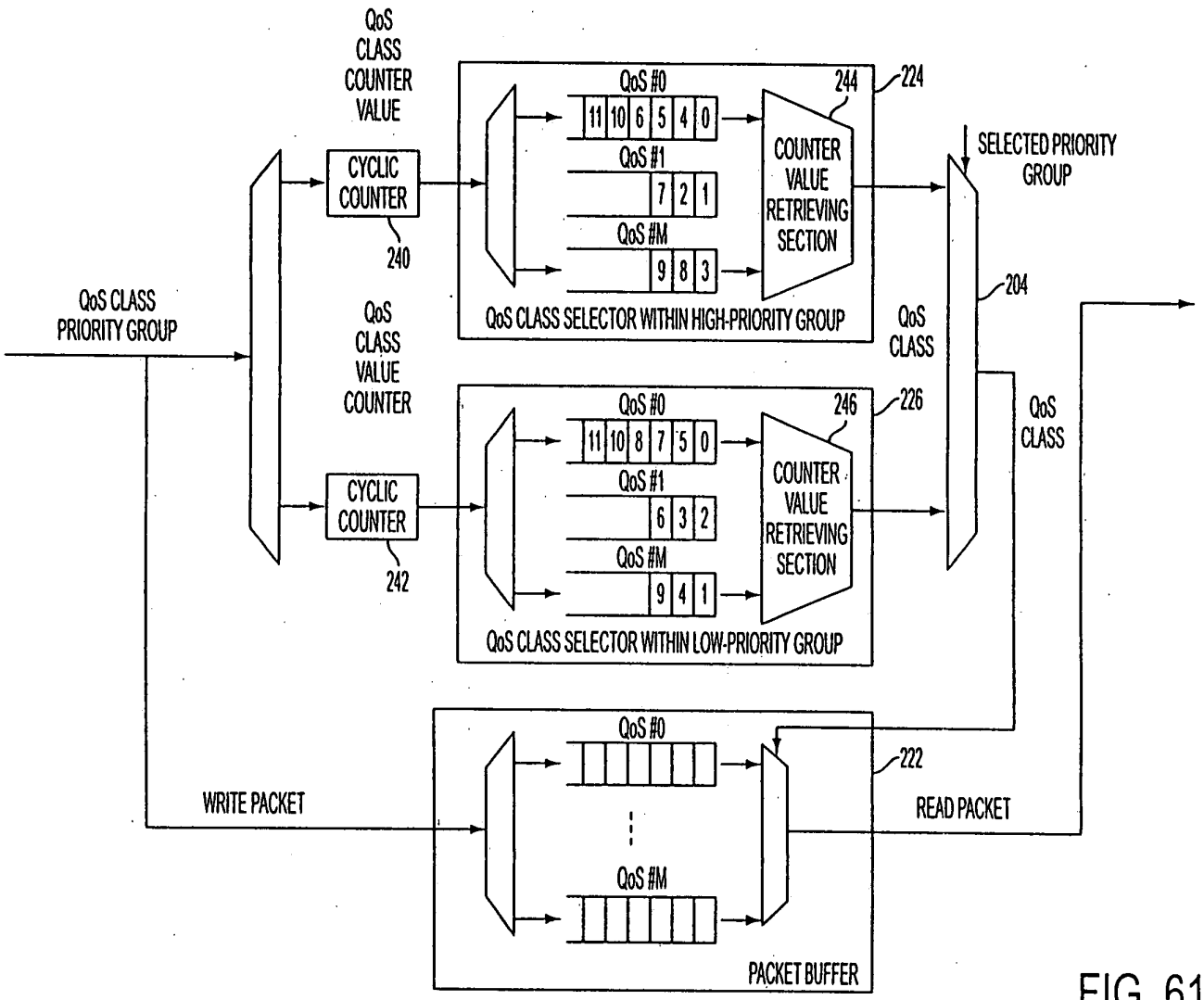


FIG. 61

FOO220 922E86D

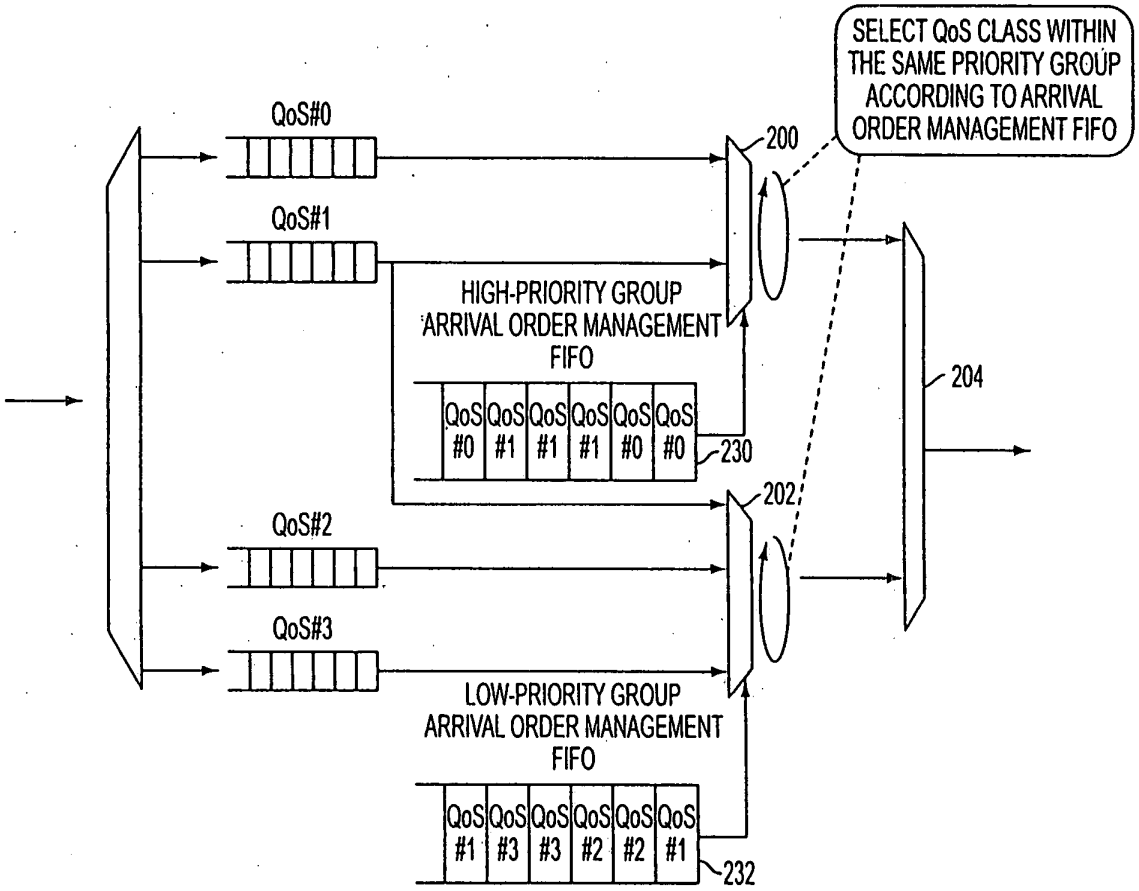


FIG. 62

T00260 922FB60

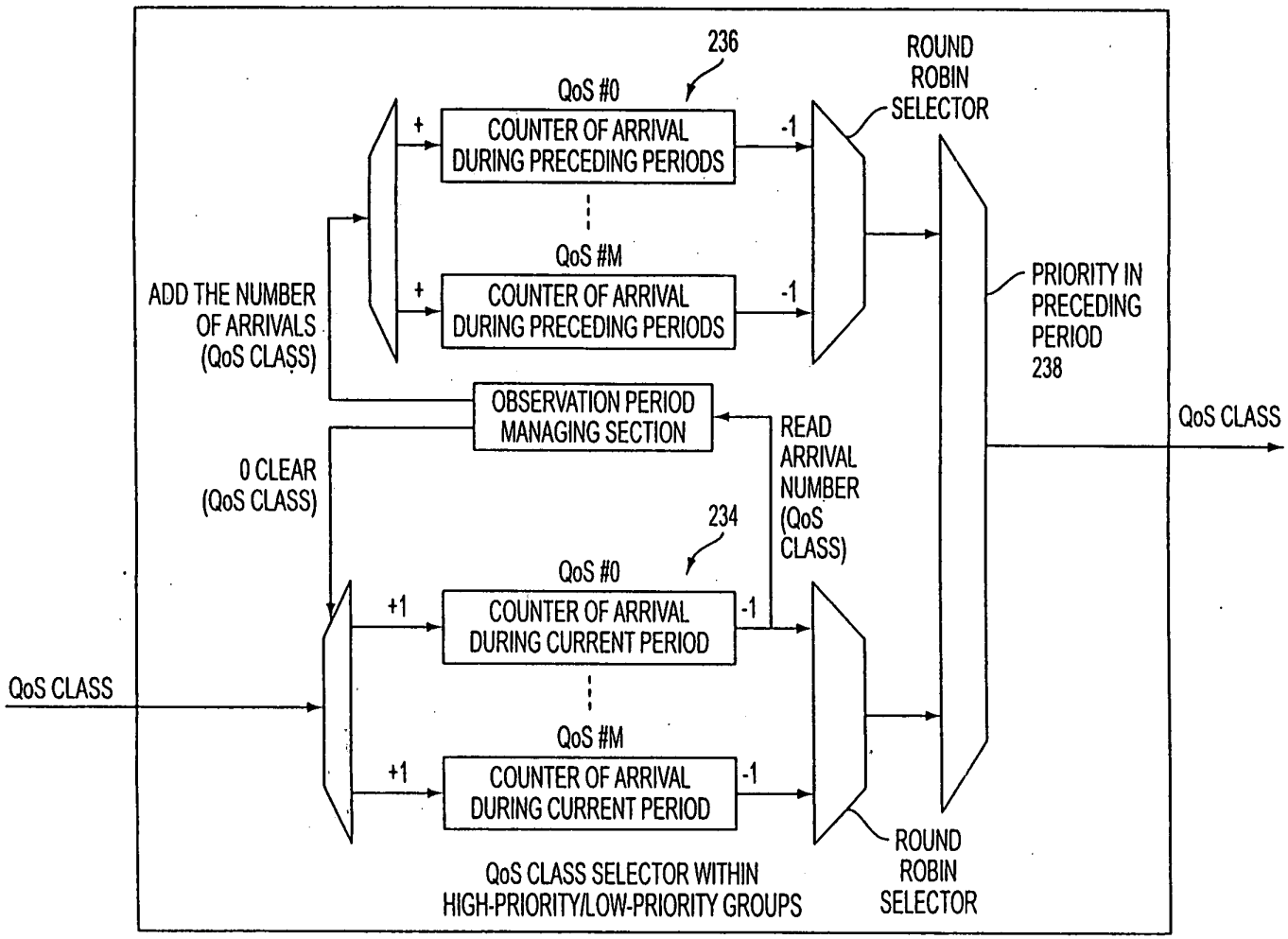


FIG. 63

T0020 92278G0

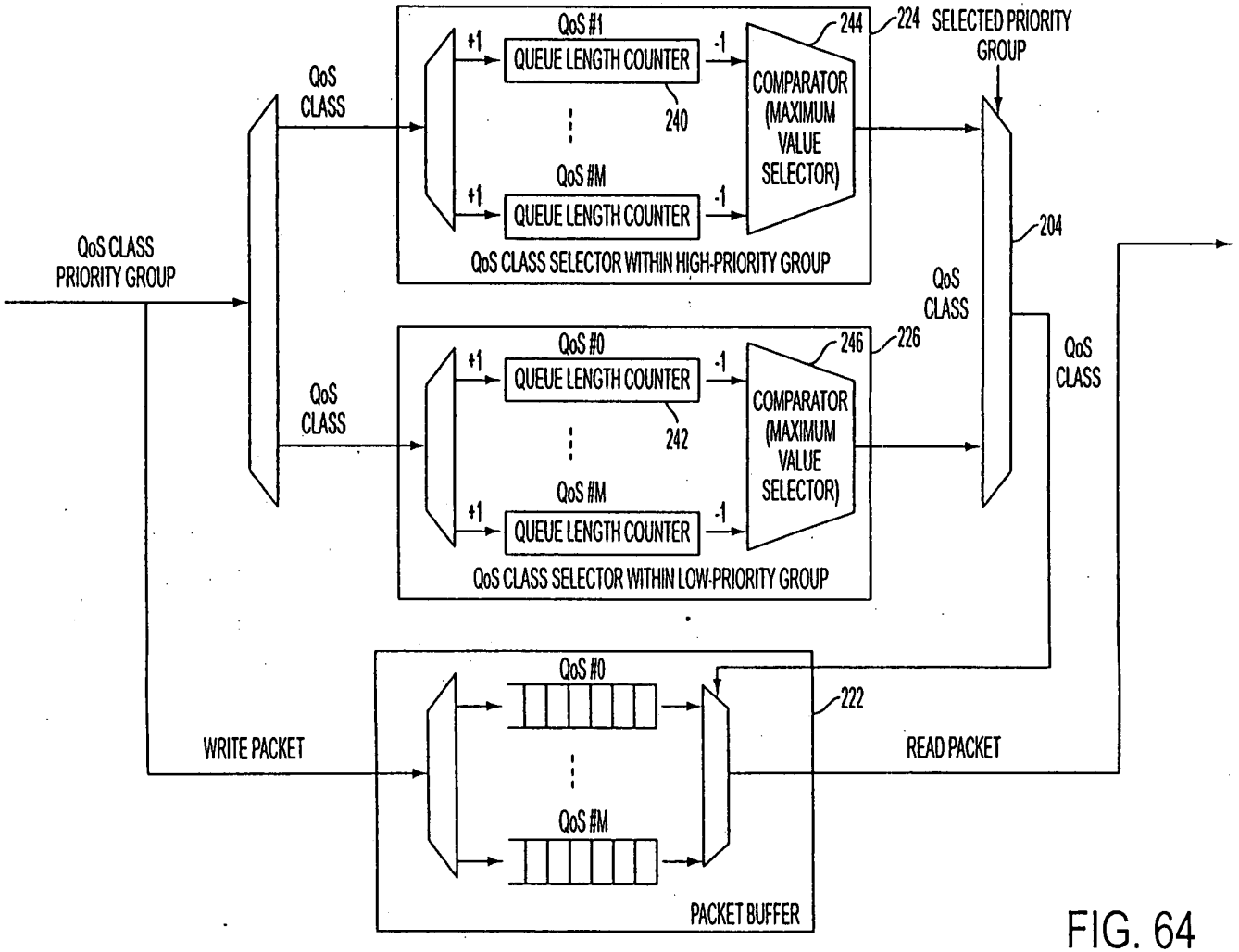


FIG. 64